Mathematics of Cognition & Intelligence

A Complete Monograph in Twelve Section s

A Complete and Explicit Guide · Every derivation unfolded
Every symbol earned · Every idea motivated

"To ask what mind is, is to ask what it means to infer.
Mathematics does not dissolve this question.
It transforms it into a landscape of structures precise enough to argue about."

Contents

0.1 Editorial Note 1

I Section **I — Probability, Belief, and Bayesian Inference** **2**

0.2 1. The Unreasonable Naturalness of Uncertainty 3

0.3 2. Probability Spaces and Kolmogorov's Axioms 3

0.3.1 2.1 The Measure-Theoretic Foundation 3

0.3.2 2.2 Random Variables and Distributions 4

0.3.3 2.3 Conditional Probability: The Heart of Inference 5

0.4 3. Bayes' Theorem: The Algebra of Belief Revision 5

0.4.1 3.1 The Theorem and Its Meaning 5

0.4.2 3.2 The Dutch Book Argument: Why Bayesian Updating Is Uniquely Rational . 6

0.4.3 3.3 Hierarchical Bayesian Models 7

0.5 4c. Practical MCMC: Diagnostics and Reparametrisation 7

0.5.1 4c.1 Divergent Transitions and Pathological Geometry 7

0.5.2 4c.2 Thinning, Burn-In, and Effective Sample Size 8

0.5.3 4c.3 Advanced MCMC: Replica Exchange and Parallel Tempering 8

0.6 3d. Bayesian Predictive Distributions and Model Comparison 9

0.6.1 3d.1 The Posterior Predictive Distribution 9

0.6.2 3d.2 Bayesian Model Comparison and Bayes Factors 10

0.6.3 3d.3 Conjugate Families: A Systematic Reference 10

0.7 3f. The Philosophy and Mathematics of Prior Selection 11

0.7.1 3f.1 Objective Priors: Jeffreys, Reference, and Maximum Entropy 11

0.7.2 3f.2 Weakly Informative and Regularising Priors 12

0.8 3e. Laplace's Method and Saddle-Point Approximations 12

0.8.1 3e.1 The Laplace Approximation 12

0.8.2 3e.2 The Saddle-Point Approximation and Cumulant Generating Functions . . . 13

0.9 4b. Markov Chain Monte Carlo 13

0.9.1 4b.1 Markov Chains and Stationary Distributions 13

0.9.2 4b.2 The Metropolis-Hastings Algorithm 14

0.9.3 4b.3 Gibbs Sampling 14

0.9.4 4b.4 Hamiltonian Monte Carlo and NUTS 15

0.9.5 4b.5 Convergence Diagnostics: $\hat{R}$ and ESS 15

0.10 3g. Hierarchical Bayesian Models: Complete Treatment 15

0.10.1 3g.1 The Eight Schools: A Canonical Worked Example 15

0.10.2 3g.2 Cognitive Hierarchical Models: Individual Differences 16
0.10.3 3g.3 Bayesian Meta-Analysis 16
0.11 5i. Kernelised Stein Discrepancy and Goodness-of-Fit Testing 17
0.11.1 5i.1 Stein Operators and Discrepancy Measures 17
0.12 14j. Counterfactual Reasoning in Cognitive Development 17
0.12.1 14j.1 Counterfactual Thinking in Children 17
0.13 13h. Mutual Information Estimation in Practice 18
0.13.1 13h.1 The MINE Estimator 18
0.14 20p. Numerical Cognition and the Approximate Number System 18
0.14.1 20p.1 Weber's Law and the ANS 18
0.15 20o. Mathematical Models of Learning Strategies 19
0.15.1 20o.1 Exploration-Exploitation in Human Cognition 19
0.16 6e. Bayesian Neural Networks: Approximation Guarantees 19
0.16.1 6e.1 The PAC-Bayesian Framework for Neural Networks 19
0.17 9l. The Free Energy Principle: A Critical Assessment 20
0.17.1 9l.1 Strengths 20
0.17.2 9l.2 Criticisms and Limitations 20
0.18 3o. Imprecise Probability and Robust Bayesian Analysis 21
0.18.1 3o.1 Credal Sets and Sensitivity Analysis 21
0.19 9k. The Mathematics of Intentions and Planning 21
0.19.1 9k.1 Hierarchical Planning and Goal Decomposition 21
0.20 13g. Information in Neural Circuits: A Synthesis 22
0.20.1 13g.1 From Shannon to Neurons: Bridging Theory and Biology 22
0.21 26j. Open Problems in Mathematical Cognitive Science 22
0.21.1 26j.1 Unsolved Mathematical Problems 22
0.22 4d. Bayesian Nonparametric Survival Analysis and Cognitive Aging 23
0.22.1 4d.1 Survival Analysis and the Hazard Function 23
0.22.2 4d.2 The Cox Proportional Hazards Model 24
0.23 18h. Causality and Interventional Probabilistic Programming 24
0.23.1 18h.1 Causal Probabilistic Programs 24
0.24 22i. Mathematical Linguistics: Formal Language Theory 24
0.24.1 22i.1 The Chomsky Hierarchy 24
0.24.2 22i.2 Probabilistic Context-Free Grammars 25
0.25 3n. The Bernstein-von Mises Theorem 25
0.25.1 3n.1 Bayesian Asymptotics 25
0.25.2 3n.2 Non-parametric Bernstein-von Mises 26
0.26 7k. Sparse Gaussian Processes: Inducing Point Methods 26
0.26.1 7k.1 The Inducing Point Framework 26
0.27 11k. Consciousness and Integrated Information: Experimental Predictions 27
0.27.1 11k.1 Testable Predictions of IIT 27
0.28 22j. The Logical Structure of Mathematical Proof 27
0.28.1 22j.1 Proof Theory and the Curry-Howard Isomorphism 27
0.29 11j. The Exclusion Postulate: Mathematical Consequences 28
0.29.1 11j.1 The Exclusion Problem 28
0.30 20n. Cognitive Maps and the Hippocampus 28

0.30.1 20n.1 O'Keefe and Nadel's Cognitive Map Theory . . . 28
0.30.2 20n.2 Path Integration and Dead Reckoning . . . 28
0.31 26i. Ecological Psychology and the Mathematics of Affordances . . . 29
0.31.1 26i.1 Affordances as Information . . . 29
0.31.2 26i.2 Motor System and Affordance Perception . . . 29
0.32 3m. Sequential Monte Carlo and Particle Filters . . . 30
0.32.1 3m.1 The Particle Filter for Online Bayesian Filtering . . . 30
0.32.2 3m.2 Rao-Blackwellised Particle Filters . . . 30
0.33 3l. Exchangeability in Practice: Bayesian Nonparametrics . . . 31
0.33.1 3l.1 Exchangeable Sequences and the Pólya Urn . . . 31
0.33.2 3l.2 Exchangeable Partitions: the Ewens Sampling Formula . . . 31
0.34 3k. A Complete Bayesian Cognitive Model: The Rational Learner . . . 32
0.34.1 3k.1 Model Architecture . . . 32
0.34.2 3k.2 Posterior and Generalisation . . . 32
0.34.3 3k.3 Extensions: The Size Principle and PAC Learning . . . 33
0.35 3j. Bayesian Hypothesis Testing vs Frequentist Testing . . . 33
0.35.1 3j.1 The p-value and Its Limitations . . . 33
0.35.2 3j.2 The Bayes Factor: Properties and Computation . . . 34
0.35.3 3j.3 Sequential Bayesian Testing: The Sequential Bayes Factor . . . 34
0.36 3i. Information Geometry of Learning and Memory . . . 34
0.36.1 3i.1 Natural Gradient Descent . . . 34
0.36.2 3i.2 Geodesics and Optimal Transport in Cognitive Change . . . 35
0.37 3h. Sequential Bayesian Updating and the Kalman Filter . . . 36
0.37.1 3h.1 Online Bayesian Updating . . . 36
0.37.2 3h.2 State-Space Models and the Kalman Filter . . . 36
0.37.3 3h.3 Extensions: Extended and Unscented Kalman Filters . . . 37
0.38 3b. Exponential Families and Sufficient Statistics . . . 37
0.38.1 3b.1 Sufficient Statistics and Fisher's Factorisation Theorem . . . 37
0.38.2 3b.2 The Exponential Family . . . 38
0.38.3 3b.3 De Finetti's Representation Theorem . . . 39
0.39 3c. Information Geometry: The Riemannian Geometry of Statistics . . . 40
0.39.1 3c.1 The Fisher-Rao Metric . . . 40
0.39.2 3c.2 Dual Connections and the α-Geometry . . . 40

II Section **II — Deep Bayesian Inference** **42**
0.40 4. The Intractability of Exact Inference . . . 43
0.41 5. Variational Inference . . . 43
0.41.1 5.1 The Variational Problem . . . 43
0.41.2 5.1b The ELBO: Complete Derivation . . . 44
0.41.3 5.2 The Mean-Field Approximation . . . 44
0.41.4 5.3 Normalising Flows: Expressive Variational Families . . . 45
0.41.5 5.4 The Variational Autoencoder . . . 45
0.42 5c. The Expectation-Maximisation Algorithm . . . 46
0.42.1 5c.1 Incomplete Data and Latent Variables . . . 46

0.42.2 5c.2 The Gaussian Mixture Model 47
0.42.3 5c.3 EM as Coordinate Ascent in the ELBO 47
0.43 5h. Expectation Propagation: Theory and Cognitive Applications 48
0.43.1 5h.1 Expectation Propagation Algorithm 48
0.43.2 5h.2 EP for Robust Regression 48
0.44 14i. Influence Diagrams and Decision-Theoretic Reasoning 49
0.44.1 14i.1 Influence Diagrams 49
0.44.2 14i.2 POMDPs as Influence Diagrams 49
0.45 16i. Modal Logic and Epistemic Accessibility 49
0.45.1 16i.1 Normal Modal Logics 49
0.45.2 16i.2 Doxastic Logic and Belief 50
0.46 22h. Formal Theories of Discourse 50
0.46.1 22h.1 Discourse Representation Theory 50
0.47 5g. Variational EM and Bayesian Model Selection 51
0.47.1 5g.1 The Evidence Lower Bound as a Model Selection Criterion 51
0.47.2 5g.2 Variational EM for Mixture Models 51
0.48 5f. Automatic Differentiation Variational Inference 52
0.48.1 5f.1 The ADVI Framework 52
0.49 5e. Variational Autoencoders: Full Mathematical Treatment 53
0.49.1 5e.1 The VAE Generative Model 53
0.49.2 5e.2 β-VAE and Disentanglement 53
0.49.3 5e.3 Hierarchical VAEs and VQ-VAE 54
0.50 6d. Bayesian Deep Learning 55
0.50.1 6d.1 The Bayesian Neural Network 55
0.50.2 6d.2 The Laplace Approximation for BNNs 55
0.50.3 6d.3 Monte Carlo Dropout as Approximate BNN 56
0.51 6c. Gaussian Process Regression: Complete Worked Example 56
0.51.1 6c.1 A Concrete Regression Problem 56
0.51.2 6c.2 Hyperparameter Learning via Marginal Likelihood 57
0.51.3 6c.3 Numerical Example 57
0.52 6b. Dirichlet Processes and Bayesian Nonparametrics 57
0.52.1 6b.1 The Dirichlet Process as a Distribution over Distributions 57
0.52.2 6b.2 The Chinese Restaurant Process and Clustering 58
0.52.3 6b.3 Gaussian Processes as Dirichlet Process Limits 58
0.53 5d. Stein Variational Gradient Descent 59
0.53.1 5d.1 Kernelised Stein Operators 59
0.53.2 5d.2 Black-Box Variational Inference 60
0.54 5b. Advanced Variational Methods 60
0.54.1 5b.1 The Importance-Weighted Autoencoder 60
0.54.2 5b.2 Stochastic Variational Inference 61
0.54.3 5b.3 Normalising Flows: Proofs and Properties 62

III Section III — Gaussian Processes and Probabilistic Function Spaces 63
0.55 6. Functions as Random Variables 64

0.56 7. Kernels and the Geometry of Beliefs 64
0.56.1 7.1 Positive Semi-Definiteness 64
0.56.2 7.2 Standard Kernels 64
0.56.3 7.3 GP Regression: Exact Posterior Inference 65
0.57 7b. Sparse Gaussian Processes 66
0.57.1 7b.1 The Computational Bottleneck 66
0.57.2 7b.2 GP Classification 66
0.57.3 7b.3 The Neural Network Gaussian Process 67
0.58 7c. Spectral Theory of Kernels 67
0.58.1 7c.1 Mercer's Theorem 67
0.58.2 7c.2 Bochner's Theorem and Spectral Densities 68
0.58.3 7c.3 Multi-Output and Multi-Task Gaussian Processes 69

IV Section **IV — Active Inference and the Free-Energy Principle** **70**
0.59 8. The Imperative to Minimise Surprise 71
0.60 9. Variational Free Energy: Formal Development 71
0.60.1 9.1 Surprise and Its Approximation 71
0.60.2 9.2 Perception, Action, and Learning as Free-Energy Minimisation 72
0.60.3 9.3 Predictive Coding: The Neural Implementation 72
0.61 7e. Gaussian Process Classification: Complete Treatment 73
0.61.1 7e.1 Laplace Approximation for GP Classification 73
0.61.2 7e.2 Expectation Propagation for GP Classification 74
0.62 7j. Reproducing Kernel Hilbert Spaces and Kernel Methods 74
0.62.1 7j.1 The Representer Theorem 74
0.62.2 7j.2 Support Vector Machines as Maximum Margin Classifiers 75
0.63 7i. Gaussian Processes in Cognitive Neuroscience 75
0.63.1 7i.1 GP Decoding of Neural Population Activity 75
0.63.2 7i.2 GP Smoothing of fMRI BOLD Signals 76
0.63.3 7i.3 Representational Similarity Analysis as a Kernel Method 76
0.64 7h. The Neural Tangent Kernel and Infinite-Width Networks 76
0.64.1 7h.1 The NTK at Initialization 76
0.64.2 7h.2 The NTK as a GP Kernel 77
0.64.3 7h.3 Finite-Width Corrections and Bayesian Deep Learning 77
0.65 7g. Bayesian Quadrature and Monte Carlo Integration 78
0.65.1 7g.1 Bayesian Quadrature 78
0.65.2 7g.2 Maximum Mean Discrepancy and Kernel Methods for Distributions 78
0.66 7f. Deep Kernel Learning and Neural Architecture Priors 79
0.66.1 7f.1 Deep Kernel Learning 79
0.66.2 7f.2 GP Prior over Neural Architecture Space 79
0.67 7d. Gaussian Processes for Temporal Cognition 80
0.67.1 7d.1 GP Models of Learning Curves 80
0.67.2 7d.2 GP-SSM: Gaussian Process State-Space Models 80
0.67.3 7d.3 Kernel Hyperparameter Learning and Model Selection 81
0.68 9j. Active Learning and Experimental Design 81

0.68.1 9j.1 Optimal Experimental Design as Active Inference . . . 81
0.68.2 9j.2 BOED with Neural Networks . . . 82
0.69 9i. Variational Free Energy: Biological Interpretations . . . 82
0.69.1 9i.1 Free Energy and Surprise . . . 82
0.69.2 9i.2 Precision and Synaptic Gain . . . 83
0.69.3 9i.3 Interoception and Homeostasis as Active Inference . . . 83
0.70 9h. Continuous-Time Active Inference . . . 84
0.70.1 9h.1 Generalised Coordinates and Continuous-Time Inference . . . 84
0.70.2 9h.2 Filtering as Free Energy Minimisation . . . 84
0.70.3 9h.3 Action as Proprioceptive Prediction Error Minimisation . . . 85
0.71 9g. Perception as Inference: Bayesian Brain Hypothesis . . . 85
0.71.1 9g.1 The Bayesian Brain Hypothesis . . . 85
0.71.2 9g.2 Predictive Coding: Derivation from First Principles . . . 86
0.71.3 9g.3 Explaining Away and Explaining Too Much . . . 86
0.72 9f. Discrete Active Inference: A Worked Example . . . 87
0.72.1 9f.1 The T-Maze Task . . . 87
0.72.2 9f.2 Policy Evaluation via Expected Free Energy . . . 87
0.72.3 9f.3 Belief Updating During Execution . . . 88
0.73 9b. POMDPs and Active Inference: The Formal Connection . . . 88
0.73.1 9b.1 Partially Observable Markov Decision Processes . . . 88
0.73.2 9b.2 Active Inference as Deep POMDP . . . 89
0.73.3 9b.3 The Free Energy of the World . . . 89
0.74 9c. Markov Blankets, Self-Organisation, and Life . . . 90
0.74.1 9c.1 The Markov Blanket Formalism . . . 90
0.74.2 9c.2 Active Inference and Reinforcement Learning . . . 90

V Section **V — Integrated Information Theory** **92**
0.75 10. The Hard Problem and the Mathematical Turn . . . 93
0.76 11. Integrated Information: The Formal Structure . . . 93
0.76.1 11.1 Cause-Effect Structures . . . 93
0.76.2 11.2 Integrated Information Φ . . . 94
0.76.3 11.3 Mathematical Properties of Φ . . . 94
0.76.4 11.4 IIT and the Geometry of Qualia . . . 95
0.77 9d. Message Passing and the Neural Implementation of Inference . . . 95
0.77.1 9d.1 Belief Propagation as Predictive Coding . . . 95
0.77.2 9d.2 Precision and Attention . . . 96
0.78 9e. Variational Message Passing and Free-Energy Minimisation in Neural Circuits . . . 96
0.78.1 9e.1 Variational Message Passing . . . 96
0.78.2 9e.2 Spike-and-Slab Priors and Synaptic Plasticity . . . 97
0.79 11b. Computational Complexity of Φ and Approximations . . . 97
0.79.1 11b.1 The #P-Hardness of Φ . . . 97
0.79.2 11b.2 Empirical Tests and Predictions . . . 98
0.79.3 11b.3 Critiques and Competing Theories . . . 98
0.80 11c. IIT 4.0: Refined Axioms and Updated Formalism . . . 99

0.80.1 11c.1 From IIT 3.0 to IIT 4.0 . . . 99
0.80.2 11c.2 A Worked Example: The XOR Gate . . . 100
0.80.3 11c.3 Phenomenal Structure and Qualia Space . . . 100

VI Section **VI — Information Theory of Mind** **102**
0.81 12. Shannon Entropy and Cognitive Uncertainty . . . 103
0.82 13. Algorithmic Information Theory and Kolmogorov Complexity . . . 103
0.83 12e. Information, Entropy, and the Physics of Cognition . . . 104
0.83.1 12e.1 Maxwell's Demon and Landauer's Principle . . . 104
0.83.2 12e.2 Free Energy and the Helmholtz Machine . . . 104
0.83.3 12e.3 Negentropy and the Free-Energy of Living Systems . . . 105
0.84 13f. Entropy Production and Cognitive Irreversibility . . . 105
0.84.1 13f.1 Entropy Production in Stochastic Systems . . . 105
0.84.2 13f.2 Mutual Information and Thermodynamic Efficiency . . . 106
0.85 13e. Causal Information Theory . . . 106
0.85.1 13e.1 Interventional Mutual Information . . . 106
0.85.2 13e.2 Transfer Entropy and Directed Information . . . 107
0.86 13d. Place Cells, Grid Cells, and the Neural Code for Space . . . 107
0.86.1 13d.1 Place Cells and Rate Maps . . . 107
0.86.2 13d.2 Grid Cells and Hexagonal Grids . . . 108
0.86.3 13d.3 Successor Representation and Predictive Maps . . . 108
0.87 13c. Algorithmic Models of Cognitive Biases . . . 109
0.87.1 13c.1 Representativeness and Kolmogorov Complexity . . . 109
0.87.2 13c.2 The Availability Heuristic and Sampling Biases . . . 109
0.87.3 13c.3 The Anchoring Effect as Insufficient Bayesian Updating . . . 109
0.88 13b. Information Theory and the Binding Problem . . . 110
0.88.1 13b.1 The Binding Problem . . . 110
0.88.2 13b.2 Rate-Distortion Theory and Efficient Coding . . . 110
0.88.3 13b.3 Kolmogorov Complexity and Solomonoff Induction . . . 111
0.89 11i. Psychophysics of Consciousness: Mathematical Experiments . . . 112
0.89.1 11i.1 The Threshold for Conscious Perception . . . 112
0.89.2 11i.2 Neural Correlates of Consciousness: Granger Cause and Effect . . . 112
0.90 11h. Network Neuroscience and Integration Measures . . . 112
0.90.1 11h.1 Functional Connectivity and Effective Connectivity . . . 112
0.90.2 11h.2 Graph-Theoretic Measures of Neural Integration . . . 113
0.91 11g. Measure-Theoretic Foundations of Integrated Information . . . 113
0.91.1 11g.1 Cause-Effect Repertoires as Probability Distributions . . . 113
0.91.2 11g.2 ϕ as an Optimal Transport Distance . . . 114
0.91.3 11g.3 The Maximum of Small Φ: Why the System Matters . . . 114
0.92 11f. Free Energy Minimisation Across Biological Scales . . . 115
0.92.1 11f.1 Cellular Free Energy Minimisation . . . 115
0.92.2 11f.2 Developmental Neuroscience as Active Inference . . . 115
0.93 11e. Empirical Measures of Consciousness . . . 116
0.93.1 11e.1 The Perturbational Complexity Index . . . 116

0.93.2 11e.2 Quantitative Neurophenomenology 117
0.94 11d. IIT and Physicalism: The Mathematical Bridge 117
0.94.1 11d.1 Identity Theory and Mathematical Characterisation 117
0.94.2 11d.2 IIT's Axiomatic Derivation in Full Detail 118
0.94.3 11d.3 Relationship to Mutual Information and Its Failure 118
0.95 12b. Neural Information Theory 119
0.95.1 12b.1 The Capacity of a Spiking Neuron 119
0.95.2 12b.2 Rate Codes versus Temporal Codes 120
0.95.3 12b.3 Minimum Description Length: A Complete Treatment 120

VII Section **VII — Probabilistic Reasoning and Uncertainty** **122**
0.96 14. Graphical Models: Representing Conditional Independence 123
0.96.1 14.1 Bayesian Networks 123
0.96.2 14.2 Causal Inference: Beyond Correlation 123
0.97 12c. Population Codes and Fisher Information 124
0.97.1 12c.1 Neural Tuning Curves and the Population Vector 124
0.97.2 12c.2 The Fisher Information Bound on Population Coding 124
0.97.3 12c.3 Redundancy and Correlation in Neural Populations 125
0.98 12d. Kolmogorov Complexity: Complete Theory and Cognitive Applications 125
0.98.1 12d.1 Kolmogorov Complexity: Definitions and Invariance 125
0.98.2 12d.2 Algorithmic Mutual Information and the Minimum Description Length Theorem 126
0.98.3 12d.3 Minimum Message Length and Human Category Learning 126
0.99 14b. Belief Propagation and Junction Tree Algorithm 127
0.99.1 14b.1 The Sum-Product Algorithm 127
0.99.2 14b.2 The Junction Tree Algorithm 128
0.100 14f. Causal Inference in Cognitive Neuroscience 128
0.100.1 14f.1 The Problem of Reverse Inference 128
0.100.2 14f.2 Granger Causality and its Limitations 129
0.100.3 14f.3 TMS as a Causal Intervention 130
0.101 14d. Variable Elimination and the Complexity of Inference 130
0.101.1 14d.1 Variable Elimination 130
0.101.2 14d.2 Approximate Inference: Loopy Belief Propagation 131
0.102 14h. Variational Inference for Causal Models 131
0.102.1 14h.1 Structural Causal Models and Posterior Inference 131
0.102.2 14h.2 Amortised Causal Inference 132
0.103 14g. Graph Neural Networks and Relational Reasoning 132
0.103.1 14g.1 Graph Neural Networks as Probabilistic Inference 132
0.103.2 14g.2 Relational Reasoning in Cognitive Science 133
0.104 14e. D-Separation, Markov Blankets, and Conditional Independence 133
0.104.1 14e.1 D-Separation: Complete Algorithm 133
0.104.2 14e.2 Faithfulness and Causal Discovery 134
0.105 14c. The Do-Calculus: Complete Rules and Applications 135
0.105.1 14c.1 The Three Rules of the Do-Calculus 135

0.105.2 14c.2 The Identification Algorithm . . . 135
0.105.3 14c.3 The Ladder of Causation . . . 136
0.105.4 14c.4 Cognitive Development and Causal Learning . . . 136

VIII Section **VIII — Non-Classical Logics and Defeasible Reasoning** **137**
0.10615. The Defeasibility of Rational Inference . . . 138
0.10716. Default Logic and Non-Monotonic Reasoning . . . 138
0.107.1 16.1 Default Rules . . . 138
0.107.2 16.2 Modal Epistemic Logic . . . 138
0.107.3 16.3 Paraconsistent Logic and Belief Revision . . . 139
0.10816h. Formal Argumentation Theory . . . 139
0.108.1 16h.1 Dung's Abstract Argumentation Frameworks . . . 139
0.108.2 16h.2 Probabilistic Argumentation . . . 140
0.10916g. Complexity of Nonmonotonic Reasoning . . . 140
0.109.1 16g.1 Complexity Classes for Nonmonotonic Inference . . . 140
0.109.2 16g.2 Approximate Nonmonotonic Reasoning . . . 141
0.11016f. Formal Epistemology: Belief Merging and Social Learning . . . 141
0.110.1 16f.1 Belief Merging Operators . . . 141
0.110.2 16f.2 Social Learning and Opinion Dynamics . . . 142
0.11116e. Default Logic: Complete Formal Treatment . . . 143
0.111.1 16e.1 Reiter's Default Logic . . . 143
0.111.2 16e.2 The Fixed-Point Equations . . . 143
0.111.3 16e.3 Cognitive Applications: Typicality and Inheritance . . . 144
0.11216b. AGM Belief Revision: The Postulates and Grove's Theorem . . . 144
0.112.1 16b.1 The Six AGM Postulates for Revision . . . 144
0.112.2 16b.2 Ranked Entailment and Spohn's Ordinal Conditional Functions . . . 145
0.11316c. Dynamic Epistemic Logic . . . 145
0.113.1 16c.1 Public Announcement Logic . . . 145
0.113.2 16c.2 Action Models and Full DEL . . . 146
0.113.3 16c.3 Conditional Logic and Non-Monotonic Consequence Relations . . . 147

IX Section **IX — Probabilistic Programming** **148**
0.11417. Programs as Models . . . 149
0.11518. Measure-Theoretic Semantics . . . 149
0.115.1 18.1 Inference Algorithms for Probabilistic Programs . . . 149
0.11616d. Quantum Logic and Paraconsistent Reasoning . . . 150
0.116.1 16d.1 The Birkhoff-von Neumann Lattice . . . 150
0.116.2 16d.2 Paraconsistent Logic . . . 150
0.116.3 16d.3 Probabilistic Logic and Markov Logic Networks . . . 151
0.11718d. Continuation-Passing Semantics and Computational Effects . . . 151
0.117.1 18d.1 The CPS Transform . . . 151
0.117.2 18d.2 Trace-Based Inference: Sequential Monte Carlo . . . 152
0.11818b. The Church Language and Probabilistic Lambda Calculus . . . 153
0.118.1 18b.1 Church as a Universal Generative Language . . . 153

0.118.2 18b.2 WebPPL: Probabilistic Programming in JavaScript . . . 154
0.118.3 18b.3 Denotational Semantics: The Probability Monad . . . 154
0.11918g. Probabilistic Graphical Models as Cognitive Architectures . . . 155
0.119.1 18g.1 Cognitive Architectures as Factor Graphs . . . 155
0.119.2 18g.2 Probabilistic Soft Logic . . . 155
0.12018f. Measure Theory and Probabilistic Program Semantics . . . 156
0.120.1 18f.1 The Radón-Nikodym Theorem and Conditioning . . . 156
0.120.2 18f.2 The Lebesgue-Stieltjes Integral and Mixed Distributions . . . 156
0.120.3 18f.3 Importance Weights and the Likelihood Weighting Algorithm . . . 157
0.12118e. Type Theory, Dependent Types, and Program Verification . . . 157
0.121.1 18e.1 Simply Typed Lambda Calculus . . . 157
0.121.2 18e.2 Dependent Types and the Calculus of Constructions . . . 158
0.121.3 18e.3 Probabilistic Types and the Probabilistic Calculus of Constructions . . . 158
0.12218c. Advanced Inference in Probabilistic Programs . . . 159
0.122.1 18c.1 Inference Compilation . . . 159
0.122.2 18c.2 Non-Parametric Bayesian Models as Probabilistic Programs . . . 159
0.122.3 18c.3 Stan and Practical Probabilistic Programming . . . 160

X Section **X — Cognitive Architectures: Mathematical Foundations** **162**
0.12319. What Is a Cognitive Architecture? . . . 163
0.12420. ACT-R: Activation, Memory, and Rational Analysis . . . 163
0.124.1 20.1 Rational Analysis . . . 163
0.124.2 20.2 Predictive Processing as Cognitive Architecture . . . 164
0.12520b. SOAR: Productions, Problem Spaces, and Chunking . . . 164
0.125.1 20b.1 The SOAR Architecture . . . 164
0.125.2 20b.2 Mathematical Cognitive Architectures: Timing and Reaction Times . . . 165
0.12620c. Global Workspace Theory: A Mathematical Formalisation . . . 166
0.126.1 20c.1 The Workspace Architecture . . . 166
0.126.2 20c.2 Mathematical Analysis of Workspace Dynamics . . . 166
0.126.3 20c.3 Predictive Processing Architecture: Neural Implementation . . . 167
0.12720d. Mathematical Models of Reaction Time and Decision Making . . . 167
0.127.1 20d.1 Sequential Probability Ratio Tests . . . 167
0.127.2 20d.2 Race Models and Independence . . . 168

XI Section **XI — Language, Formalism, and Categorical Cognition** **169**
0.12821. The Compositionality Problem in Cognition . . . 170
0.12922. Montague Grammar: Intensional Logic and Compositional Semantics . . . 170
0.129.1 22.1 The Typed Lambda Calculus . . . 170
0.129.2 22.2 Compositional Semantics: The Homomorphism Principle . . . 170
0.13023. Categorical Compositional Distributional Semantics . . . 171
0.130.1 23.1 Monoidal Categories . . . 171
0.130.2 23.2 DisCoCat: Syntax Meets Semantics Categorically . . . 171
0.13120g. Mathematical Models of Attention . . . 172
0.131.1 20g.1 Signal Detection Theory for Attention . . . 172

0.131.2 20g.2 The Attentional Blink: Temporal Bottleneck Models . . . 172
0.131.3 20g.3 Spatial Attention: the Biased Competition Model . . . 173
0.13220e. Neural Networks as Cognitive Architectures . . . 173
0.132.1 20e.1 The Universal Approximation Theorem . . . 173
0.132.2 20e.2 Transformer Architecture as Approximate Bayesian Inference . . . 174
0.13320m. Mathematical Models of Cognitive Development . . . 175
0.133.1 20m.1 Stochastic Grammar Induction . . . 175
0.133.2 20m.2 Theory Learning and Conceptual Development . . . 175
0.13420l. Working Memory: Capacity and Forgetting . . . 176
0.134.1 20l.1 The Magic Number Seven . . . 176
0.134.2 20l.2 Mathematical Models of Forgetting . . . 176
0.13520k. Embodied Simulation and Motor Cognition . . . 176
0.135.1 20k.1 Mirror Neurons and Action Understanding . . . 176
0.135.2 20k.2 Corollary Discharge and Sensory Attenuation . . . 177
0.13620j. Reinforcement Learning: Mathematical Taxonomy . . . 177
0.136.1 20j.1 The RL Framework . . . 177
0.136.2 20j.2 Model-Free vs Model-Based RL . . . 178
0.136.3 20j.3 Deep Reinforcement Learning . . . 178
0.13720i. The Mathematics of Metacognition . . . 179
0.137.1 20i.1 Confidence as Posterior Probability . . . 179
0.137.2 20i.2 The Metacognitive Read-Out Model . . . 179
0.137.3 20i.3 Hierarchical Metacognition and the Self-Model . . . 180
0.13820h. Optimal Control and the Mathematics of Motor Learning . . . 180
0.138.1 20h.1 Linear Quadratic Regulators . . . 180
0.138.2 20h.2 Stochastic Optimal Control and Hamilton-Jacobi-Bellman . . . 181
0.138.3 20h.3 Reward-Modulated Hebbian Plasticity and TD Learning . . . 181
0.13920f. ACT-R: The Mathematical Core . . . 182
0.139.1 20f.1 The Declarative Memory Module: Activation Equations . . . 182
0.139.2 20f.2 Retrieval Time and Threshold . . . 182
0.139.3 20f.3 The Procedural Module and Production Rules . . . 183
0.14022b. Game Semantics and Ludics . . . 183
0.140.1 22b.1 Game Semantics for Logic . . . 183
0.140.2 22b.2 Pregroup Grammars: Completeness and Computational Complexity . . . 184
0.140.3 22b.3 Distributional Compositional Semantics: Extensions . . . 184
0.14122g. Formal Pragmatics and Game Theory of Language . . . 185
0.141.1 22g.1 Gricean Maxims as Bayesian Inference . . . 185
0.141.2 22g.2 Signalling Games and the Evolution of Meaning . . . 185
0.14222f. Spatial Language and Conceptual Metaphor . . . 186
0.142.1 22f.1 The Geometry of Spatial Prepositions . . . 186
0.142.2 22f.2 Conceptual Metaphor Theory and Its Formalisation . . . 186
0.14322e. Compositionality and the Architecture of Meaning . . . 187
0.143.1 22e.1 The Principle of Compositionality: Mathematical Formulation . . . 187
0.143.2 22e.2 Tensor Product Representations . . . 187
0.143.3 22e.3 Cognitive Architectures for Language: From RNNS to Transformers . . . 188
0.14422d. Montague Semantics: Complete Treatment . . . 189

0.144.1 22d.1 Intensional Logic and Possible Worlds . . . 189
0.144.2 22d.2 Quantifier Raising and LF . . . 189
0.144.3 22d.3 Dynamic Semantics and Discourse . . . 190
0.14522c. Type-Theoretic Semantics and Proof-Theoretic Meaning . . . 190
0.145.1 22c.1 Propositions as Types: The Curry-Howard Correspondence . . . 190
0.145.2 22c.2 Homotopy Type Theory and Cognitive Representations . . . 191
0.145.3 22c.3 Vector Space Models: Empirical Validation . . . 191

XII Section **XII — Embodied Cognition and Mathematical Limits** **193**
0.14624. The Body in the Mind: Beyond the Representationalist Paradigm . . . 194
0.14725b. Impossibility Theorems and the Limits of Formalisation . . . 194
0.147.1 25b.1 Gödel's Incompleteness and the Limits of Mathematical Cognition . . . 194
0.147.2 25b.2 The Frame Problem: Formal Analysis . . . 195
0.147.3 25b.3 Coupled Dynamical Systems: Stability and Bifurcations . . . 195
0.14823c. Experimental Tests of Compositional Semantics . . . 196
0.148.1 23c.1 Semantic Similarity: Benchmarks and Results . . . 196
0.148.2 23c.2 Entailment and Recognising Textual Entailment . . . 196
0.14923b. Category Theory of Meaning: Monoidal Structures . . . 197
0.149.1 23b.1 Monoidal Categories and Tensor Products . . . 197
0.149.2 23b.2 The DisCoCat Functor as a Semantics . . . 197
0.149.3 23b.3 Lambek Calculus and Type-Logical Grammar . . . 198
0.15025c. The Geometry of Action and Sensorimotor Space . . . 199
0.150.1 25c.1 Motor Synergies and Dimensionality Reduction . . . 199
0.150.2 25c.2 Riemannian Geometry of Musculoskeletal Systems . . . 199
0.150.3 25c.3 Enactivism and Mathematical Limits of Representation . . . 200
0.15125. Dynamical Systems: The Mathematics of Coupling . . . 201
0.151.1 25.1 Continuous Dynamical Systems . . . 201
0.151.2 25.2 Agent-Environment Coupling . . . 201
0.15226. Mathematical Limits: What Formal Cognition Cannot Capture . . . 201
0.152.1 26.1 The Grounding Problem . . . 201
0.152.2 26.2 The Limits of the Bayesian Framework . . . 202
0.15326b. The Mathematics of Semantic Spaces . . . 203
0.153.1 26b.1 Gärdenfors' Conceptual Spaces . . . 203
0.153.2 26b.2 The Information Geometry of Conceptual Change . . . 203
0.153.3 26b.3 Persistence Homology and the Topology of Neural Representations . . . 204
0.15426h. Embodied Intelligence and Mathematical Frontiers . . . 204
0.154.1 26h.1 Morphological Computation . . . 204
0.154.2 26h.2 The Computational Limits of Learning . . . 205
0.15526g. Phase Transitions and the Emergence of Consciousness . . . 205
0.155.1 26g.1 First-Order Transitions and Conscious Access . . . 205
0.155.2 26g.2 Critical Brain Dynamics and Consciousness . . . 206
0.15626f. Integrated Mathematical Perspectives on Cognition . . . 206
0.156.1 26f.1 The Bayesian Brain as a Unified Framework . . . 206
0.156.2 26f.2 Information Geometry as the Meta-Framework . . . 207

0.156.3 26f.3 Open Mathematical Problems in Mathematical Cognitive Science 207
0.15726e. Self-Organisation, Emergence, and Complexity in Cognition 208
0.157.1 26e.1 Complex Adaptive Systems and the Edge of Chaos 208
0.157.2 26e.2 Renormalisation Group and Multi-Scale Cognition 208
0.157.3 26e.3 Integrated Information Geometry and Consciousness 209
0.15826d. Quantum Cognition: A Mathematical Assessment 209
0.158.1 26d.1 The Quantum Probability Framework 209
0.158.2 26d.2 The Conjunction Fallacy and Quantum Interference 210
0.158.3 26d.3 Limitations and Assessment 210
0.15926c. Neural Field Theory: Mathematics of Brain Dynamics 211
0.159.1 26c.1 Wilson-Cowan Neural Field Equations 211
0.159.2 26c.2 Attractor Dynamics and Working Memory 211
0.159.3 26c.3 Oscillations, Synchrony, and Binding by Synchrony 212
0.16027. Open Frontiers: Theorems at the Edge of Understanding 212
0.1613p. Robust Bayesian Estimation and M-Estimators 213
0.161.1 3p.1 M-Estimators as Approximate MAP 213
0.16226k. The Hard Problem of Consciousness: Mathematical Perspectives 214
0.162.1 26k.1 Mathematical Approaches to the Explanatory Gap 214
0.1633q. Bayesian Experimental Psychology: Model Fitting and Identifiability 214
0.163.1 3q.1 The Identifiability Problem in Cognitive Models 214
0.163.2 3q.2 Approximate Bayesian Computation 215
0.163.3 3q.3 Posterior Predictive Model Checking 216
0.16412c. Fisher Information and the Geometry of Neural Coding 216
0.164.1 12c.1 Population Codes and Fisher Information 216
0.164.2 12c.2 The Geometry of Tuning Curves 216
0.164.3 12c.3 Information-Limiting Correlations 217
0.16513d. Grid Cells, Entorhinal Cortex, and the Mathematics of Spatial Codes 217
0.165.1 13d.1 The Grid Cell Code 217
0.165.2 13d.2 Capacity and Decoding of Grid Codes 218
0.165.3 13d.3 Path Integration and Attractor Dynamics 218
0.16614k. Reinforcement Learning: Mathematical Foundations 219
0.166.1 14k.1 Markov Decision Processes 219
0.166.2 14k.2 Temporal Difference Learning 219
0.166.3 14k.3 The Dopamine-TD Correspondence 220
0.166.4 14k.4 Deep Reinforcement Learning 220
0.16717e. Stochastic Processes in Cognition: A Synthesis 221
0.167.1 17e.1 The Ornstein-Uhlenbeck Process and Evidence Accumulation 221
0.167.2 17e.2 Diffusion Models: Analytical Results 221
0.167.3 17e.3 Point Processes and Spike Train Statistics 222
0.16818i. Causal Discovery Algorithms: Complete Treatment 222
0.168.1 18i.1 The PC Algorithm 222
0.168.2 18i.2 Score-Based Methods: GES and FGES 223
0.168.3 18i.3 Functional Causal Models and Non-Gaussianity 223
0.16919g. Statistical Learning Theory: Generalisation Bounds 224
0.169.1 19g.1 VC Dimension and Uniform Convergence 224

0.169.2 19g.2 Margin Bounds and Deep Networks . . . 224
0.17020q. Cognitive Complexity and Algorithmic Information . . . 225
0.170.1 20q.1 Kolmogorov Complexity and Cognitive Simplicity . . . 225
0.170.2 20q.2 Descriptive Complexity and Cognitive Grammars . . . 225
0.170.3 20q.3 Minimum Description Length in Practice . . . 226
0.17121g. Topological Data Analysis and Cognitive Representations . . . 226
0.171.1 21g.1 Persistent Homology . . . 226
0.171.2 21g.2 Topological Analysis of Neural Representations . . . 227
0.171.3 21g.3 Mapper Algorithm for Brain Networks . . . 227
0.17222k. Probabilistic Semantics and Meaning Uncertainty . . . 228
0.172.1 22k.1 Probabilistic Type Theory . . . 228
0.172.2 22k.2 Lexical Semantics: Distributional Models . . . 228
0.172.3 22k.3 Transformer Architecture and Attention . . . 229
0.17323e. Embodied Mathematics: Enactive Cognition and Dynamical Systems . . . 229
0.173.1 23e.1 Enactivism and Sensorimotor Contingencies . . . 229
0.173.2 23e.2 Dynamical Systems Theory of Coordination . . . 230
0.173.3 23e.3 Thermodynamics of Cognition and Landauer's Principle . . . 231
0.17424d. Category Theory and Cognition: A Deeper Look . . . 231
0.174.1 24d.1 Adjunctions and Cognitive Processes . . . 231
0.174.2 24d.2 Monads and Computational Effects . . . 232
0.174.3 24d.3 Sheaves and Contextuality . . . 232
0.17525e. Mathematical Models of Social Cognition . . . 233
0.175.1 25e.1 Theory of Mind and Higher-Order Beliefs . . . 233
0.175.2 25e.2 Game Theory and Strategic Cognition . . . 233
0.175.3 25e.3 Cultural Evolution and Mathematical Models of Transmission . . . 234
0.176Appendix D: Mathematical Prerequisites and Background . . . 234
0.176.1 D.1 Linear Algebra . . . 234
0.176.2 D.2 Measure Theory and Integration . . . 235
0.176.3 D.3 Functional Analysis . . . 235
0.176.4 D.4 Differential Geometry . . . 235
0.177Appendix A: Core Theorems of the Series . . . 235
0.178Appendix B: Connections Across the Series . . . 236
0.179Appendix C: Glossary of Mathematical Terms . . . 237
0.18028. Continual Learning and Catastrophic Forgetting . . . 238
0.180.1 28.1 The Stability-Plasticity Dilemma . . . 238
0.180.2 28.2 Elastic Weight Consolidation . . . 238
0.180.3 28.3 Progressive Networks and Architecture-Based Approaches . . . 239
0.180.4 28.4 Hippocampal Replay and Complementary Learning Systems . . . 239
0.18129. Neuro-Symbolic Integration: Bridging Continuous and Discrete Cognition . . . 240
0.181.1 29.1 The Integration Problem . . . 240
0.181.2 29.2 Probabilistic Soft Logic . . . 240
0.181.3 29.3 Neural Theorem Proving . . . 240
0.181.4 29.4 DeepProbLog and Probabilistic Logic Neural Networks . . . 241
0.18230. Mathematical Models of Emotion and Affective Cognition . . . 241
0.182.1 30.1 Emotion as Interoceptive Inference . . . 241

0.182.2 30.2 Dimensional Models: Valence and Arousal 242
0.182.3 30.3 Appraisal Theory: A Multi-Dimensional Computational Model 242
0.182.4 30.4 Emotion Regulation as Optimal Control 242
0.18331. AI Alignment: Mathematical Foundations 243
0.183.1 31.1 The Alignment Problem Stated Formally 243
0.183.2 31.2 Inverse Reinforcement Learning 243
0.183.3 31.3 Cooperative AI and Principal-Agent Problems 244
0.183.4 31.4 Reward Modelling and Scalable Oversight 244
0.183.5 31.5 Formal Verification and Specification 245
Bibliography 246

0.1 Editorial Note

This monograph collects all twelve Section s of the *Mathematics of Cognition & Intelligence* series into a single continuous work. The Section s were originally conceived as standalone texts; read together, they trace a single progressive argument — that a mathematically serious account of mind and intelligence requires the simultaneous deployment of probability theory, information theory, differential geometry, the calculus of variations, category theory, logic, and the theory of neural computation. Each framework illuminates a facet that the others leave in shadow; none is sufficient alone.

The series is the theoretical complement to the computational perspectives of *Matneural*. Where that series covers the engineering and algorithmic side of intelligent systems, this one covers the mathematical structure of the mind itself: what it means to hold a belief, to minimise surprise, to integrate information, to reason under uncertainty, to act in a body embedded in a world. No existing series connects all these threads in a unified and rigorous way. The ambition of this one is exactly that.

Section	Title	Central Framework
I	Probability, Belief, and Bayesian Inference	Probability · Bayes · Hierarchical models
II	Deep Bayesian Inference	Variational Bayes · Normalising flows
III	Gaussian Processes and Probabilistic Function Spaces	Kernels · GP regression · GP classification
IV	Active Inference and the Free-Energy Principle	Variational free energy · Friston
V	Integrated Information Theory	Φ · Cause-effect structures · Tononi
VI	Information Theory of Mind	Shannon · MDL · Kolmogorov complexity
VII	Probabilistic Reasoning and Uncertainty	Graphical models · Causal inference
VIII	Non-Classical Logics and Defeasible Reasoning	Modal · Paraconsistent · Default logic
IX	Probabilistic Programming	Measure-theoretic semantics · Inference
X	Cognitive Architectures: Mathematical Foundations	ACT-R · SOAR · Predictive processing
XI	Language, Formalism, and Categorical Cognition	Montague · Monoidal categories · Syntax
XII	Embodied Cognition and Mathematical Limits	Dynamical systems · Enactivism · Limits

The reader is assumed to be comfortable with calculus, linear algebra, and basic probability at the undergraduate level. Every concept beyond that starting point is developed from first principles: no result is invoked without proof or at least a carefully motivated proof sketch, and no symbol is used without definition. The style is deliberately unhurried. Ideas that took decades for their inventors to clarify are given the space they deserve.

A remark on notation. We write $P(\cdot)$ and $p(\cdot)$ interchangeably for probability measures and densities, disambiguating by context. Random variables are set in Roman type; their realisations in lower case. Expectations are written $\mathbb{E}_{p(x)}[f(x)]$ or $\mathbb{E}[f(x)]$ when the distribution is clear. Mutual information is $\mathrm{I}(X;Y)$; differential entropy is $\mathrm{H}(X)$; KL divergence is $\mathrm{KL}(p\|q)$.

Section *I*

Section I — Probability, Belief, and Bayesian Inference

0.2 1. The Unreasonable Naturalness of Uncertainty

There is an old fantasy in the history of science: the fantasy of complete knowledge. Laplace articulated it most clearly when he imagined a demon who knew the position and momentum of every particle in the universe and could therefore compute, in principle, every future and every past state of affairs. In this picture, uncertainty is not a feature of the world but a deficiency of the observer. It is epistemic, not ontic — a gap in knowledge, not a gap in nature.

This picture has been challenged from many directions. Quantum mechanics introduced genuine irreducible randomness at the physical level. Chaos theory showed that deterministic systems can be unpredictable in practice due to sensitivity to initial conditions. And the study of complex cognitive systems has made it clear that an agent embedded in a world it can only partially observe must, as a matter of logical necessity, reason in terms of degrees of belief rather than certainties.

This series begins with probability theory because probability is the mathematics of uncertain belief. It is not a collection of tricks for computing with dice. It is a comprehensive formal framework for representing what an agent knows and does not know, and for updating that knowledge as evidence arrives. When an agent is Bayesian — when it represents its beliefs as probability distributions and updates them according to Bayes' rule — it is behaving in the uniquely coherent way permitted by the axioms of probability. Any other way of combining evidence leads, in provable and precise senses, to contradiction.

The deep fact that makes Bayesian inference interesting as a model of cognition is this: a brain that predicted the world perfectly would have nothing to do. It is precisely because brains must operate under uncertainty — because they must infer hidden causes from noisy sensory data — that probabilistic inference is not merely a computational convenience but a structural feature of intelligent processing. The mathematics we develop in this and the following Section s is, we will argue, the mathematics of mind itself.

0.3 2. Probability Spaces and Kolmogorov's Axioms

0.3.1 2.1 The Measure-Theoretic Foundation

The modern foundation of probability theory rests on measure theory, codified by Andrey Kolmogorov in 1933. The central object is a **probability space** $(\Omega, \mathcal{F}, P)$, consisting of three components.

The **sample space** Ω is the set of all possible outcomes. For a coin toss, $\Omega = \{H, T\}$. For the height of a randomly selected adult, $\Omega = \mathbb{R}_{>0}$. For the trajectory of a particle, Ω is a space of continuous functions. The sample space is the most primitive object: it is the catalogue of everything that could happen.

The **σ-algebra** $\mathcal{F}$ is a collection of subsets of Ω, called **events**, satisfying three closure conditions: (i) $\Omega \in \mathcal{F}$; (ii) if $A \in \mathcal{F}$ then $\Omega \setminus A \in \mathcal{F}$ (closed under complementation); (iii) if $A_1, A_2, \ldots \in \mathcal{F}$ then $\bigcup_{i=1}^{\infty} A_i \in \mathcal{F}$ (closed under countable unions). The σ-algebra is the collection of events to which we are willing to assign probabilities.

The **probability measure** $P : \mathcal{F} \to [0, 1]$ assigns to each event a number in $[0, 1]$ satisfying two axioms: (i) $P(\Omega) = 1$ (the certain event has probability 1); (ii) for any countable collection of mutually disjoint

events $A_1, A_2, \ldots \in \mathcal{F}$ with $A_i \cap A_j = \emptyset$ for $i \neq j$,

$$P\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} P(A_i).$$

This is **countable additivity** (or σ-additivity). Together, these are Kolmogorov's axioms.

From these sparse axioms, all of probability theory is derived. The probability of the empty event is zero: $P(\emptyset) = 0$, because $\Omega \cup \emptyset = \Omega$ and $P(\Omega) = P(\Omega) + P(\emptyset)$ by disjoint additivity, so $P(\emptyset) = 0$. Finite additivity follows: $P(A \cup B) = P(A) + P(B)$ whenever $A \cap B = \emptyset$, as a special case of countable additivity. The complement rule: $P(A^c) = 1 - P(A)$, because A and A^c are disjoint with $A \cup A^c = \Omega$. The inclusion-exclusion principle: $P(A \cup B) = P(A) + P(B) - P(A \cap B)$, because we can write $A \cup B = A \cup (B \setminus A)$ with the latter disjoint and then note $P(B) = P(A \cap B) + P(B \setminus A)$.

The measure-theoretic formulation is not mere abstraction. It is essential for handling continuous random variables, infinite sequences of events, and the delicate limit theorems (laws of large numbers, central limit theorem) that underlie essentially all of statistical inference.

0.3.2 2.2 Random Variables and Distributions

A **random variable** $X : \Omega \to \mathbb{R}$ is a measurable function from the sample space to the real numbers. Measurability means that for every Borel set $B \subseteq \mathbb{R}$, the pre-image $X^{-1}(B) = \{\omega \in \Omega : X(\omega) \in B\}$ belongs to $\mathcal{F}$, so that we can meaningfully speak of the probability $P(X \in B)$.

The **distribution** (or **law**) of X is the probability measure P_X on $\mathbb{R}$ defined by $P_X(B) = P(X^{-1}(B))$. The distribution packages all probabilistic information about X into a measure on the real line, discarding the sample space. In practice, we work almost entirely with distributions rather than sample spaces.

A discrete random variable takes values in a countable set $\{x_1, x_2, \ldots\}$. Its distribution is specified by the **probability mass function** (pmf) $p(x_i) = P(X = x_i)$, which satisfies $p(x_i) \geq 0$ and $\sum_i p(x_i) = 1$.

A continuous random variable has a **probability density function** (pdf) $p(x) \geq 0$ such that

$$P(X \in B) = \int_B p(x)\, dx, \quad \int_{-\infty}^{\infty} p(x)\, dx = 1.$$

The density $p(x)$ is not a probability: $p(x)$ can exceed 1, and $P(X = x) = 0$ for every individual point x.

Example (Gaussian distribution). The most important distribution in all of probability theory is the Gaussian (normal) distribution $X \sim \mathcal{N}(\mu, \sigma^2)$ with mean $\mu \in \mathbb{R}$ and variance $\sigma^2 > 0$, whose density is

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right).$$

The Gaussian is distinguished by three properties. First, it is the maximum-entropy distribution subject only to fixed mean and variance (we will prove this in Section VI). Second, by the Central Limit Theorem, it arises as the limiting distribution of sums of independent, identically distributed random variables with finite variance. Third, it is preserved under linear transformations and convolutions, making it the natural distribution for modelling additive noise in cognitive and neural systems.

0.3.3 2.3 Conditional Probability: The Heart of Inference

Conditional probability is the formal machinery for updating beliefs in the light of evidence. The conditional probability of event A given event B (with $P(B) > 0$) is defined as

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)}.$$

This is not a new axiom but a definition. Its justification is conceptual: if we learn that B occurred, then the effective sample space shrinks to B, and probabilities must be re-normalised within this new space. The formula is the precise expression of that re-normalisation.

Conditional probability satisfies all the Kolmogorov axioms in its second argument: $P(\cdot \mid B)$ is itself a probability measure on $\mathcal{F}$. This means all the derived rules of probability theory apply to conditional probabilities.

The **chain rule** (or product rule) follows immediately from the definition:

$$P(A \cap B) = P(A \mid B)\, P(B) = P(B \mid A)\, P(A).$$

Extending by induction:

$$P(A_1 \cap A_2 \cap \cdots \cap A_n) = P(A_1)\, P(A_2 \mid A_1)\, P(A_3 \mid A_1, A_2) \cdots P(A_n \mid A_1, \ldots, A_{n-1}).$$

This decomposition is fundamental in graphical models, where it corresponds to the factorisation of a joint distribution along a directed acyclic graph.

The **law of total probability** provides a way to compute unconditional probabilities by conditioning on a partition. If $\{B_1, B_2, \ldots\}$ is a countable partition of Ω (mutually exclusive, collectively exhaustive events with positive probability), then

$$P(A) = \sum_i P(A \mid B_i)\, P(B_i).$$

This is the mathematical expression of marginalisation: summing (or integrating) out the conditioning variable.

0.4 3. Bayes' Theorem: The Algebra of Belief Revision

0.4.1 3.1 The Theorem and Its Meaning

Bayes' theorem is a simple algebraic consequence of the product rule. From $P(A \cap B) = P(A \mid B)P(B) = P(B \mid A)P(A)$, rearranging:

$$P(A \mid B) = \frac{P(B \mid A)\, P(A)}{P(B)}.$$

In the language of statistical inference, we replace A with a hypothesis θ and B with observed data D:

$$p(\theta \mid D) = \frac{p(D \mid \theta)\, p(\theta)}{p(D)}.$$

The four components have names that encode their roles. The **prior** $p(\theta)$ is the agent's distribution over hypotheses before observing any data: it encodes prior knowledge, assumptions, and structural constraints. The **likelihood** $p(D \mid \theta)$ specifies, for each hypothesis θ, the probability of generating the observed data D: it connects the model to the observations. The **posterior** $p(\theta \mid D)$ is the agent's updated distribution over hypotheses after observing D: it is the goal of Bayesian inference. The **evidence** (or **marginal likelihood**) $p(D) = \int p(D \mid \theta)p(\theta)\, d\theta$ is the normalisation constant ensuring the posterior integrates to one.

The evidence is often computed using the law of total probability:

$$p(D) = \int p(D \mid \theta)\, p(\theta)\, d\theta.$$

For discrete hypothesis spaces, the integral becomes a sum. The evidence does not depend on θ, so for the purpose of computing the posterior's shape, we often write the **unnormalised posterior**:

$$p(\theta \mid D) \propto p(D \mid \theta)\, p(\theta).$$

The symbol $\propto$ means "proportional to as a function of θ"; the normalisation constant $p(D)$ is inferred from the requirement that the posterior integrates to one.

Example (Coin toss). Suppose we wish to estimate the bias $\theta \in [0, 1]$ of a coin (the probability of heads). We observe a sequence of n tosses with k heads. The likelihood is

$$p(D \mid \theta) = \binom{n}{k} \theta^k (1-\theta)^{n-k}.$$

A natural choice of prior is the **Beta distribution** $p(\theta) = \text{Beta}(\alpha, \beta) \propto \theta^{\alpha-1}(1-\theta)^{\beta-1}$ with hyperparameters $\alpha, \beta > 0$. The Beta distribution is the **conjugate prior** for the binomial likelihood: the posterior is in the same family as the prior. Computing:

$$p(\theta \mid D) \propto \theta^k (1-\theta)^{n-k} \cdot \theta^{\alpha-1}(1-\theta)^{\beta-1} = \theta^{(\alpha+k)-1}(1-\theta)^{(\beta+n-k)-1}.$$

This is $\text{Beta}(\alpha+k,\ \beta+n-k)$. The prior counts α and β are updated by the observed counts k and $n-k$: evidence simply increments the pseudo-counts. The posterior mean is $(\alpha+k)/(\alpha+\beta+n)$, interpolating between the prior mean $\alpha/(\alpha+\beta)$ and the maximum likelihood estimate k/n.

0.4.2 3.2 The Dutch Book Argument: Why Bayesian Updating Is Uniquely Rational

The Bayesian framework is not merely one approach to inference among many. It is, under certain assumptions about rational behaviour, the unique coherent approach. The **Dutch Book argument** makes this precise.

An agent's beliefs are **coherent** if they cannot be exploited by a clever bookmaker: there is no set of bets, each accepted by the agent at their stated probabilities, that is guaranteed to lose money regardless of the outcome. The Dutch Book theorem states that an agent's beliefs are coherent if and only if they satisfy the Kolmogorov axioms. Any system of beliefs that violates the axioms is incoherent: there exists a combination of bets that will guarantee the agent a loss.

The dynamic version, due to David Lewis, extends this to belief updating: an agent who does not update

by Bayes' rule can be made to lose money over time through a sequence of bets made before and after observing evidence. Bayesian updating is, therefore, not a heuristic but a logical necessity for any agent that aspires to coherence in its beliefs.

For cognitive science, this is a deep claim. It says that a rational mind must, in the relevant abstract sense, be a Bayesian inference machine. This is the hypothesis driving the Bayesian Brain programme that runs as a thread through Section s I–V of this series.

0.4.3 3.3 Hierarchical Bayesian Models

A single Bayes' theorem update describes learning about a single parameter from data. Real cognitive problems involve **structure at multiple levels**: learning the parameters of a model while simultaneously learning the structure of the model; inferring the properties of a particular object while inferring the statistical regularities of the class to which it belongs. **Hierarchical Bayesian models** formalise this multi-level learning.

In a two-level hierarchical model, the data D depends on parameters θ, and θ depends on hyperparameters ϕ:

$$p(D, \theta, \phi) = p(D \mid \theta)\, p(\theta \mid \phi)\, p(\phi).$$

The joint posterior is

$$p(\theta, \phi \mid D) \propto p(D \mid \theta)\, p(\theta \mid \phi)\, p(\phi).$$

The hyperparameters ϕ are not fixed in advance but are themselves inferred from the data. This is how the model learns its own inductive biases: the prior $p(\theta \mid \phi)$ is flexible, and the data shape it through ϕ.

The cognitive analogue is **meta-learning**: an agent that has learned from many previous tasks acquires a prior over parameters that is well-suited to new tasks in the same domain. The hyperparameters ϕ encode this "prior over priors." Hierarchical models are thus the formal machinery for formalising what psychologists call *learning to learn* and neuroscientists call *hyperpriors*.

Example (Hierarchical Gaussian model). Suppose we observe n data points from n different but related experiments, where experiment i produces observation $y_i \sim \mathcal{N}(\theta_i, \sigma^2)$ and the experiment-level parameters share a common prior $\theta_i \sim \mathcal{N}(\mu, \tau^2)$. The hyperparameters are (μ, τ^2). The posterior for each θ_i is:

$$\theta_i \mid y_i, \mu, \tau^2 \sim \mathcal{N}\left(\frac{\sigma^{-2} y_i + \tau^{-2}\mu}{\sigma^{-2} + \tau^{-2}},\ \frac{1}{\sigma^{-2} + \tau^{-2}}\right).$$

This is the weighted average of the individual observation y_i and the population mean μ, with weights proportional to the precisions σ^{-2} and τ^{-2}. When τ^2 is large (the prior is diffuse), the individual observation dominates. When τ^2 is small (the prior is tight), the population mean dominates. This **shrinkage** phenomenon — pulling individual estimates toward the group mean — is the signature of hierarchical inference and corresponds, cognitively, to the use of context in perception and judgement.

0.5 4c. Practical MCMC: Diagnostics and Reparametrisation

0.5.1 4c.1 Divergent Transitions and Pathological Geometry

Modern HMC samplers (NUTS in Stan) report **divergent transitions**: leapfrog trajectories that deviate significantly from a conserved Hamiltonian. A divergent transition is a warning sign that the posterior ge-

ometry is poorly conditioned in that region — the curvature changes too rapidly for the leapfrog integrator to track accurately, leading to large integration errors.

Mathematically, a divergent transition occurs when the leapfrog Hamiltonian error $|\Delta H| = |H(\theta^*, r^*) - H(\theta, r)|$ exceeds a threshold (typically $\Delta H > 1000$ in Stan). Geometrically, divergences indicate that the sampler is attempting to explore a **narrow neck** or **funnel** in the posterior: a region where the scale of the posterior changes dramatically over a short distance.

The Neal's funnel. The canonical example: $v \sim \mathcal{N}(0, 3^2)$, $x|v \sim \mathcal{N}(0, e^v)$. The joint distribution is a "funnel": for large $|v|$, x is nearly zero; for small $|v|$, x can be large. The scale of x varies by a factor of e^3 over the typical range of v. The leapfrog step size that is appropriate for large $|v|$ is far too large for small $|v|$, causing divergences.

Non-centred parametrisation resolves the funnel. Replace $x|v$ by $\tilde{x} = x/e^{v/2}$ (the standardised variable), so $\tilde{x} \sim \mathcal{N}(0, 1)$ independent of v. The reparametrised posterior $p(v, \tilde{x}) = p(v)p(\tilde{x})$ is a product of two standard Gaussians with no correlation — perfectly conditioned for HMC.

The general principle: whenever a parameter's scale depends on another parameter (hierarchical models, scale mixtures), use the non-centred parametrisation to decouple scale from location:

- Centred: $\theta_j \sim \mathcal{N}(\mu, \sigma^2) \to$ Non-centred: $\theta_j = \mu + \sigma\eta_j$, $\eta_j \sim \mathcal{N}(0, 1)$.
- Centred: $\beta_j \sim \text{Cauchy}(0, \tau) \to$ Non-centred: $\beta_j = \tau z_j$, $z_j \sim \text{Cauchy}(0, 1)$.

0.5.2 4c.2 Thinning, Burn-In, and Effective Sample Size

Burn-in: the initial phase of an MCMC chain where the chain has not yet reached the stationary distribution. Samples from the burn-in phase should be discarded. The required burn-in is related to the **mixing time** τ_{mix}: the number of steps for the chain to reach within total variation distance ϵ of the stationary distribution from any starting point.

For a geometrically ergodic chain (which includes HMC under mild conditions), the burn-in needed for ϵ accuracy is $O(\tau_{\text{mix}} \log(1/\epsilon))$ steps. In practice, $\hat{R}$ (Section 4b.5) is used to detect non-convergence: run multiple chains and discard samples until $\hat{R} < 1.01$.

Thinning (keeping every k-th sample) reduces autocorrelation but does not improve the statistical efficiency of estimates: T/k thinned samples contain the same information as T unthinned samples, since the effective sample size ESS $\approx T/(1 + 2\sum_{j=1}^{\infty} \rho_j)$ is determined by the autocorrelation, not the thinning. Thinning only saves storage, not computation. The recommendation: do not thin; use all samples and report ESS.

Trace plots visualise the chain's history: a well-mixed chain shows a "fuzzy caterpillar" pattern with no trends or periodic behaviour. A poorly-mixed chain shows slow drift, sticking (long runs in one region), or periodic oscillations. For multivariate posteriors, trace plots should be examined for each marginal, and pairwise scatter plots of posterior samples should be examined for multi-modality.

0.5.3 4c.3 Advanced MCMC: Replica Exchange and Parallel Tempering

For multimodal posteriors (multiple well-separated modes), standard MCMC chains get trapped in individual modes and cannot cross the low-probability regions between them. **Parallel tempering** (or replica

exchange MCMC) runs K chains at different "temperatures" $T_1 < T_2 < \cdots < T_K$:

$$p_k(\theta) \propto p(\theta|D)^{1/T_k}.$$

Hot chains ($T_k \gg 1$) have flattened posteriors and can easily jump between modes. Cold chains ($T_k = 1$) sample the true posterior. At regular intervals, adjacent chains exchange states with probability:

$$P(\text{swap}_{k,k+1}) = \min\left(1, \frac{p_{k+1}(\theta_k)p_k(\theta_{k+1})}{p_k(\theta_k)p_{k+1}(\theta_{k+1})}\right),$$

satisfying detailed balance. Hot chains explore globally; cold chains refine locally; swaps transfer globally-discovered modes to the cold chain.

For cognitive modelling, parallel tempering is essential when the posterior has genuine multimodality (e.g., mixture models where the labelling is symmetric, or models with multiple regimes). The temperature schedule T_k should be chosen so that adjacent chains have acceptance rates ≈ 0.23–0.38, analogous to the optimal acceptance rate for scalar Metropolis.

0.6 3d. Bayesian Predictive Distributions and Model Comparison

0.6.1 3d.1 The Posterior Predictive Distribution

One of the most important applications of Bayesian inference is **prediction**: given past data $D = (x_1, \ldots, x_n)$, what is the probability of a new observation x_{n+1}? The answer is the **posterior predictive distribution**:

$$p(x_{n+1} \mid D) = \int p(x_{n+1} \mid \theta)p(\theta \mid D)\, d\theta.$$

This is not a plug-in estimate $p(x_{n+1} \mid \hat{\theta})$ (which uses the MLE or MAP estimate and ignores posterior uncertainty) but the genuine Bayesian average over all possible parameter values, weighted by their posterior probabilities. The posterior predictive **marginalises out** the parameter uncertainty.

Example (Gaussian predictive). For a normal-normal conjugate model with known variance: $x_i \sim \mathcal{N}(\mu, \sigma^2)$, $\mu \sim \mathcal{N}(\mu_0, \sigma_0^2)$. The posterior is $\mu|D \sim \mathcal{N}(\mu_n, \sigma_n^2)$ where:

$$\mu_n = \frac{\sigma^{-2}\sum_i x_i + \sigma_0^{-2}\mu_0}{\sigma^{-2}n + \sigma_0^{-2}}, \qquad \sigma_n^{-2} = n\sigma^{-2} + \sigma_0^{-2}.$$

The predictive distribution for x_{n+1} is:

$$x_{n+1} \mid D \sim \mathcal{N}\big(\mu_n,\ \sigma^2 + \sigma_n^2\big)\,.$$

The predictive variance $\sigma^2 + \sigma_n^2$ has two components: σ^2 (irreducible aleatoric uncertainty, the noise in each observation) and σ_n^2 (epistemic uncertainty about μ, which shrinks as $n \to \infty$). As $n \to \infty$, $\sigma_n^2 \to 0$ and the predictive distribution concentrates on the true Gaussian $\mathcal{N}(\mu_{\text{true}}, \sigma^2)$.

The posterior predictive correctly *propagates* parameter uncertainty into predictive uncertainty. In contrast, the plug-in estimate $p(x_{n+1}|\hat{\mu}) = \mathcal{N}(x_{n+1}; \hat{\mu}, \sigma^2)$ understates uncertainty, especially when n is small and $\hat{\mu}$ is poorly determined.

0.6.2 3d.2 Bayesian Model Comparison and Bayes Factors

Given two models M_0 and M_1, the Bayesian approach to model comparison uses the **Bayes factor**:

$$K = \frac{p(D \mid M_1)}{p(D \mid M_0)} = \frac{\int p(D|\theta_1, M_1) p(\theta_1|M_1)\, d\theta_1}{\int p(D|\theta_0, M_0) p(\theta_0|M_0)\, d\theta_0}.$$

The Bayes factor is the ratio of marginal likelihoods (evidences). Combined with prior model probabilities $p(M_0)$ and $p(M_1) = 1 - p(M_0)$, the posterior odds are:

$$\frac{p(M_1|D)}{p(M_0|D)} = K \cdot \frac{p(M_1)}{p(M_0)}.$$

Jeffreys' (1961) calibration table for Bayes factors: $K < 1$ (evidence for M_0); $1 \le K < 3$ (anecdotal evidence for M_1); $3 \le K < 10$ (moderate evidence); $10 \le K < 30$ (strong evidence); $K \ge 100$ (decisive evidence for M_1).

Savage-Dickey density ratio. For nested models $M_0 \subset M_1$ (where M_0 is obtained from M_1 by fixing $\theta_1 = \theta_0$), the Bayes factor simplifies to the ratio of the marginal posterior density to the prior density at the nested parameter value:

$$K = \frac{p(D|M_0)}{p(D|M_1)} = \frac{p(\theta_1 = \theta_0 \mid D, M_1)}{p(\theta_1 = \theta_0 \mid M_1)}.$$

This is especially useful in cognitive modelling: to test whether a parameter (e.g., a learning rate) is zero versus non-zero, compute the posterior density at zero divided by the prior density at zero.

Bayesian Information Criterion (BIC). The Laplace approximation to the marginal likelihood gives:

$$\log p(D|M) \approx \log p(D|\hat{\theta}_{\text{MAP}}) - \frac{k}{2} \log n + O(1),$$

where k is the number of parameters. The BIC $= -2 \log p(D|\hat{\theta}) + k \log n$ approximates $-2 \log p(D|M)$ up to constants, penalising model complexity. BIC is consistent: it selects the true model as $n \to \infty$ if the true model is in the comparison set. It is asymptotically equivalent to MDL (Section 13b) and to Bayes factors with a specific reference prior.

0.6.3 3d.3 Conjugate Families: A Systematic Reference

Likelihood	Prior	Posterior	Predictive
Bernoulli(θ)	Beta(α, β)	Beta($\alpha + k, \beta + n - k$)	Beta-Binomial
Binomial(n, θ)	Beta(α, β)	Beta($\alpha + k, \beta + n - k$)	Beta-Binomial
Poisson(λ)	Gamma(α, β)	Gamma($\alpha + \sum x_i, \beta + n$)	Neg-Binomial
Normal(μ, σ^2)	Normal(μ_0, σ_0^2)	Normal(μ_n, σ_n^2)	Normal($\mu_n, \sigma^2 + \sigma_n^2$)
Normal(μ, σ^2)	Normal-InvGamma	Normal-InvGamma	Student-t
Exponential(λ)	Gamma(α, β)	Gamma($\alpha + n, \beta + \sum x_i$)	Lomax
Multinomial($n, \boldsymbol{\theta}$)	Dirichlet($\boldsymbol{\alpha}$)	Dirichlet($\boldsymbol{\alpha} + \mathbf{n}$)	Dirichlet-Multinomial

Each conjugate pair satisfies: the posterior is in the same family as the prior, with updated hyperparameters that are linear functions of the sufficient statistics of the data. This property makes Bayesian

updating computationally trivial: no integration required, just addition of pseudo-counts or adjustment of hyperparameters.

The deeper mathematical reason for conjugacy is the exponential family structure (Section 3b): the conjugate prior to an exponential family likelihood is itself an exponential family, because the product $p(\theta|D) \propto p(D|\theta)p(\theta)$ of two exponential-form functions is again exponential-form.

0.7 3f. The Philosophy and Mathematics of Prior Selection

0.7.1 3f.1 Objective Priors: Jeffreys, Reference, and Maximum Entropy

The choice of prior is the most controversial aspect of Bayesian inference. Three principled approaches have been developed to select priors with minimal subjective content.

Jeffreys prior. The Jeffreys prior $p_J(\theta) \propto \sqrt{\det \mathcal{I}(\theta)}$ is invariant under reparametrisation: if $\phi = g(\theta)$, then the Jeffreys prior for ϕ is the pushforward of the Jeffreys prior for θ under g. This invariance is the key property: the prior does not favour any particular parametrisation of the model.

For a Bernoulli likelihood $p(k|\theta) = \theta^k(1-\theta)^{1-k}$: $\mathcal{I}(\theta) = 1/(\theta(1-\theta))$, so $p_J(\theta) \propto \theta^{-1/2}(1-\theta)^{-1/2} = \text{Beta}(1/2, 1/2)$. This prior places more mass near 0 and 1 (the extreme success/failure probabilities), reflecting the fact that the data are most informative about θ near these extremes.

For a Poisson likelihood $p(k|\lambda) = e^{-\lambda}\lambda^k/k!$: $\mathcal{I}(\lambda) = 1/\lambda$, so $p_J(\lambda) \propto \lambda^{-1/2}$: an improper prior (not normalisable) that is uniform on $\sqrt{\lambda}$. This is one of many cases where the Jeffreys prior is improper; validity of the posterior requires checking that $\int p(\lambda|D)\, d\lambda < \infty$.

Reference priors (Bernardo 1979; Berger and Bernardo 1992) select the prior that maximises the expected KL divergence between posterior and prior (the expected information gained from the experiment). Formally:

$$p_{\text{ref}} = \arg\max_{p} \mathbb{E}_{p(D)}[\text{KL}(p(\theta|D)\|p(\theta))].$$

Reference priors maximise the "information brought by the data" and are the Jeffreys prior in one dimension (but differ in higher dimensions due to nuisance parameters).

Maximum entropy priors. Given a set of moment constraints $\mathbb{E}[f_i(\theta)] = c_i$ for $i = 1, \ldots, k$, the **maximum entropy prior** is:

$$p_{\text{MaxEnt}} = \arg\max_{p:\mathbb{E}[f_i]=c_i} H(p) = \frac{1}{Z}\exp\left(-\sum_i \lambda_i f_i(\theta)\right),$$

which is an exponential family distribution with sufficient statistics $\{f_i\}$. The Lagrange multipliers λ_i are determined by the constraints. Maximum entropy selects the "flattest" prior consistent with the constraints, in the information-theoretic sense of entropy.

The maximum entropy principle (Jaynes 1957) provides a unified justification for common priors: the uniform prior is MaxEnt with no constraints; the Gaussian $\mathcal{N}(\mu, \sigma^2)$ is MaxEnt given $\mathbb{E}[\theta] = \mu$ and $\mathbb{E}[\theta^2] = \sigma^2 + \mu^2$; the exponential distribution is MaxEnt given $\mathbb{E}[\theta] = \mu$ and $\theta \geq 0$.

0.7.2 3f.2 Weakly Informative and Regularising Priors

In practice, the distinction between "objective" and "subjective" priors is less important than the distinction between **weakly informative priors** (which encode broad domain knowledge without strongly constraining the posterior) and **strongly informative priors** (which reflect confident prior beliefs).

The **half-Cauchy prior** $\sigma \sim \text{Half-Cauchy}(0, 1)$ for scale parameters is weakly informative: it assigns non-negligible probability to very large values of σ (heavy tails), preventing the prior from ruling out large effects, while still providing gentle regularisation toward zero. The Cauchy distribution has no finite moments, making it maximally robust to outliers in the prior.

The **regularised horseshoe prior** (Piironen and Vehtari 2017) for sparse regression: in a regression $y = X\beta + \epsilon$, many components of β may be zero (the sparsity assumption). The horseshoe prior:

$$\beta_j|\lambda_j, \tau \sim \mathcal{N}(0, \lambda_j^2\tau^2), \quad \lambda_j \sim \text{Half-Cauchy}(0, 1), \quad \tau \sim \text{Half-Cauchy}(0, \tau_0),$$

encourages sparsity: the global scale τ sets the overall level of shrinkage, and the local scales λ_j allow individual coefficients to escape the shrinkage (the "horseshoe" effect at $\lambda_j \approx 1$). The regularised horseshoe truncates the local scales to be bounded: $\tilde{\lambda}_j^2 = c^2\lambda_j^2/(c^2 + \tau^2\lambda_j^2)$, where c^2 is a slab variance, preventing the prior from assigning very large prior probability to extremely large coefficients.

The horseshoe prior has been shown to be **Bayes-optimal** for sparse normal-means estimation in the regime $n \to \infty$, $s/n \to 0$ (where s is the number of non-zero signals): the posterior mean under the horseshoe achieves the minimax risk for ℓ_2 estimation up to a $\log n$ factor (van der Pas et al., 2016).

0.8 3e. Laplace's Method and Saddle-Point Approximations

0.8.1 3e.1 The Laplace Approximation

When the posterior $p(\theta|D)$ is unimodal and approximately Gaussian near its mode, the **Laplace approximation** provides a closed-form Gaussian approximation that is computationally cheap and often surprisingly accurate.

Let $\hat{\theta} = \arg\max_\theta \log p(\theta|D)$ be the MAP estimate. Expanding $\log p(\theta|D)$ to second order around $\hat{\theta}$:

$$\log p(\theta|D) \approx \log p(\hat{\theta}|D) - \frac{1}{2}(\theta - \hat{\theta})^\top A(\theta - \hat{\theta}),$$

where $A = -\nabla^2 \log p(\theta|D)|_{\theta=\hat{\theta}}$ is the **observed information matrix** (negated Hessian of the log-posterior at the mode). The approximation gives:

$$p(\theta|D) \approx \mathcal{N}(\theta; \hat{\theta}, A^{-1}).$$

This is the Laplace approximation. The approximate posterior is Gaussian with mean equal to the MAP and covariance equal to the inverse observed information.

Approximation to the marginal likelihood. Integrating the Laplace approximation:

$$p(D) = \int p(D|\theta)p(\theta)\, d\theta \approx p(D|\hat{\theta})p(\hat{\theta})(2\pi)^{d/2}|A|^{-1/2},$$

where d is the dimension of θ. The log marginal likelihood is:

$$\log p(D) \approx \log p(D|\hat{\theta}) + \log p(\hat{\theta}) + \frac{d}{2}\log(2\pi) - \frac{1}{2}\log|A|.$$

This is the basis of the BIC (dropping the $\log p(\hat{\theta})$ and $\frac{d}{2}\log(2\pi)$ terms, which are $O(1)$, and noting that $|A| = O(n^d)$).

Accuracy. The Laplace approximation is accurate when: (1) the posterior is unimodal, (2) the mode is well-identified (sharp peak), and (3) the posterior is approximately symmetric. It fails badly for multi-modal posteriors (misses all but one mode), heavy-tailed posteriors (underestimates tail probability), and low-dimensional posteriors with strong non-Gaussianity.

For logistic regression with N data points and d parameters, the Laplace approximation is $O(Nd + d^3)$ (the cost of computing A and its Cholesky decomposition), compared to MCMC at $O(T \cdot Nd)$ for T samples. For large d, the $O(d^3)$ Cholesky cost dominates.

0.8.2 3e.2 The Saddle-Point Approximation and Cumulant Generating Functions

The Laplace method extends to integrals of the form $\int \exp(nf(x))\,dx$ for large n. If f has a unique maximum at x^*, the integral is dominated by the contribution near x^*:

$$\int_{\mathbb{R}} \exp(nf(x))\,dx \approx \sqrt{\frac{2\pi}{n|f''(x^*)|}}\exp(nf(x^*)).$$

The **saddlepoint approximation** for probability distributions uses the **cumulant generating function** (CGF) $K(t) = \log \mathbb{E}[e^{tX}]$. For a distribution with CGF K, the saddlepoint approximation to the density at x is:

$$\hat{p}(x) = \sqrt{\frac{1}{2\pi K''(\hat{t})}}\exp(K(\hat{t}) - \hat{t}x),$$

where $\hat{t}$ is the solution to the saddlepoint equation $K'(\hat{t}) = x$. This approximation is accurate in the tails (where Gaussian approximations fail) and is the basis for Lugannani-Rice tail probability approximations used in mathematical statistics and actuarial science.

For cognitive modelling, saddlepoint approximations provide accurate approximations to reaction time distributions (which are typically right-skewed and heavy-tailed) from drift-diffusion models, without the need for numerical simulation. The CGF of the first-passage time of a Brownian motion with drift ν and boundary a can be computed analytically:

$$K(t) = a\nu/\sigma^2 - a\sqrt{\nu^2/\sigma^4 - 2t/\sigma^2},$$

and the saddlepoint approximation to the RT density is accurate to $O(n^{-3/2})$.

0.9 4b. Markov Chain Monte Carlo

0.9.1 4b.1 Markov Chains and Stationary Distributions

When the posterior distribution $p(\theta \mid D)$ is high-dimensional and non-Gaussian, variational methods introduce systematic bias. **Markov Chain Monte Carlo** (MCMC) provides an alternative: construct a

Markov chain whose stationary distribution *is* the posterior and draw samples by running the chain.

A Markov chain on state space Θ has transition kernel $T(\theta'|\theta)$. A distribution π is **stationary** if $\int T(\theta'|\theta)\pi(d\theta) = \pi(\theta')$. The **detailed balance** condition $\pi(\theta)T(\theta'|\theta) = \pi(\theta')T(\theta|\theta')$ implies stationarity (integrate both sides over θ). For an irreducible, aperiodic chain, the ergodic theorem gives:

$$\frac{1}{T}\sum_{t=1}^{T} f(\theta^{(t)}) \xrightarrow{a.s.} \mathbb{E}_\pi[f(\theta)], \qquad \|T^t(\cdot|\theta^{(0)}) - \pi\|_{\text{TV}} \to 0.$$

0.9.2 4b.2 The Metropolis-Hastings Algorithm

The Metropolis-Hastings (MH) algorithm constructs a kernel with stationary distribution $\pi = p(\theta|D)$ using only the unnormalised posterior $\tilde{p}(\theta) = p(D|\theta)p(\theta)$:

1. Propose $\theta' \sim q(\cdot|\theta^{(t)})$.
2. Compute $\alpha = \min\left(1,\ \dfrac{\tilde{p}(\theta')\, q(\theta^{(t)}|\theta')}{\tilde{p}(\theta^{(t)})\, q(\theta'|\theta^{(t)})}\right)$.
3. Set $\theta^{(t+1)} = \theta'$ w.p. α, else $\theta^{(t+1)} = \theta^{(t)}$.

Proposition 0.9.1. The MH kernel satisfies detailed balance with respect to $\pi \propto \tilde{p}$.

Proof. WLOG $\tilde{p}(\theta')q(\theta|\theta') \le \tilde{p}(\theta)q(\theta'|\theta)$. Then $\alpha(\theta', \theta) = \tilde{p}(\theta')q(\theta|\theta')/(\tilde{p}(\theta)q(\theta'|\theta))$ and $\alpha(\theta, \theta') = 1$. Compute:

$$\pi(\theta)T(\theta'|\theta) = \tilde{p}(\theta)q(\theta'|\theta) \cdot \frac{\tilde{p}(\theta')q(\theta|\theta')}{\tilde{p}(\theta)q(\theta'|\theta)} = \tilde{p}(\theta')q(\theta|\theta') = \pi(\theta')T(\theta|\theta'). \quad \square$$

For a d-dimensional isotropic Gaussian target, Random Walk MH (with $q(\theta'|\theta) = \mathcal{N}(\theta; \theta^{(t)}, \sigma^2 I)$) has optimal acceptance rate ≈ 0.234 at $\sigma^2 \approx 2.38^2/d$, and mixing time $O(d)$.

0.9.3 4b.3 Gibbs Sampling

Gibbs sampling updates each coordinate i by drawing from its exact full conditional:

$$\theta_i^{(t+1)} \sim p(\theta_i \mid \theta_1^{(t+1)}, \ldots, \theta_{i-1}^{(t+1)}, \theta_{i+1}^{(t)}, \ldots, \theta_d^{(t)}).$$

This is MH with the full conditional as proposal, giving acceptance probability 1. For conjugate models, full conditionals are standard distributions. For Bayesian linear regression $y = X\beta + \epsilon$ with $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$, $\beta \sim \mathcal{N}(0, \tau^2 I)$, $\sigma^2 \sim$ Inv-Gamma(a_0, b_0):

$$\beta|\cdot \sim \mathcal{N}\left(\left(\frac{X^\top X}{\sigma^2} + \frac{I}{\tau^2}\right)^{-1} \frac{X^\top y}{\sigma^2},\ \left(\frac{X^\top X}{\sigma^2} + \frac{I}{\tau^2}\right)^{-1}\right),$$

$$\sigma^2|\cdot \sim \text{Inv-Gamma}\left(a_0 + \frac{n}{2},\ b_0 + \frac{\|y - X\beta\|^2}{2}\right).$$

Mixing degrades when parameters are strongly correlated; centring and non-centred parametrisations improve convergence.

0.9.4 4b.4 Hamiltonian Monte Carlo and NUTS

HMC exploits gradient information to make large, nearly-independent moves. Augment θ with momentum $r \sim \mathcal{N}(0, M)$. Define $H(\theta, r) = U(\theta) + \frac{1}{2} r^\top M^{-1} r$ where $U(\theta) = -\log \tilde{p}(\theta)$. Hamiltonian dynamics preserve H:

$$\dot{\theta} = M^{-1} r, \qquad \dot{r} = -\nabla U(\theta).$$

The Section -preserving, time-reversible **leapfrog integrator** (L steps of size ε) discretises these equations with $O(\varepsilon^2)$ energy error. The terminal state (θ^*, r^*) is accepted with probability $\min(1, \exp(-\Delta H))$. HMC's mixing time is $O(d^{1/4})$ for well-conditioned targets, vs. $O(d)$ for Random Walk MH. The **No-U-Turn Sampler** (NUTS) selects the number of leapfrog steps automatically by detecting trajectory reversal, eliminating manual tuning.

0.9.5 4b.5 Convergence Diagnostics: $\hat{R}$ and ESS

Convergence is assessed empirically. The **Gelman-Rubin** $\hat{R}$ runs m chains of length n, computing between-chain variance B and within-chain variance W:

$$\hat{R} = \sqrt{\frac{\widehat{\text{Var}}^+}{W}}, \quad \widehat{\text{Var}}^+ = \frac{n-1}{n} W + \frac{B}{n}.$$

Values $\hat{R} > 1.01$ indicate non-convergence. The **effective sample size** $\text{ESS} = N/(1+2\sum_k \rho_k)$ accounts for autocorrelation; Monte Carlo error scales as $\hat{\sigma}/\sqrt{\text{ESS}}$.

0.10 3g. Hierarchical Bayesian Models: Complete Treatment

0.10.1 3g.1 The Eight Schools: A Canonical Worked Example

The **eight schools** problem (Rubin 1981; Gelman et al., 2013) is the canonical example of hierarchical modelling. Eight schools implement a new educational programme; the estimated effect sizes y_j and their standard errors σ_j are observed:

School	y_j	σ_j
A	28	15
B	8	10
C	-3	16
D	7	11
E	-1	9
F	1	11
G	18	10
H	12	18

The hierarchical model is: $y_j \sim \mathcal{N}(\theta_j, \sigma_j^2)$ (observed), $\theta_j \sim \mathcal{N}(\mu, \tau^2)$ (latent per-school effect), $\mu \sim \mathcal{N}(0, 100^2)$, $\tau \sim$ Half-Cauchy$(0, 25)$ (hyperpriors).

The complete posterior $p(\theta_{1:8}, \mu, \tau | y_{1:8})$ is computed by MCMC. Key insights:

Partial pooling. The posterior school effects $\hat{\theta}_j$ are a compromise between the no-pooling estimates y_j

(treat each school independently) and the complete-pooling estimate $\bar{y}$ (assume all schools are identical):

$$\hat{\theta}_j \approx B_j y_j + (1 - B_j)\hat{\mu}, \qquad B_j = \frac{\sigma_j^2}{\sigma_j^2 + \tau^2}.$$

Here B_j is the **shrinkage coefficient**: how much to shrink toward the grand mean. School A ($y_A = 28$, $\sigma_A = 15$) has large noise and is shrunk substantially toward $\hat{\mu} \approx 8$; School E ($y_E = -1$, $\sigma_E = 9$) has lower noise and is shrunk less.

The role of τ. The hyperparameter τ controls the between-school variation. For $\tau \to 0$ (complete pooling), all $\theta_j \to \mu$; for $\tau \to \infty$ (no pooling), $\theta_j \to y_j$. The posterior $p(\tau|y)$ is updated from the half-Cauchy prior by the likelihood of observing the between-school variance var$(y_j - \mu)$.

Practical estimates. The posterior mode of τ is $\hat{\tau} \approx 7$ (in the original data), indicating moderate between-school variation. The posterior means are: $\hat{\theta}_A = 12$, substantially less than $y_A = 28$ (shrinkage toward the mean), but still above average. The uncertainty in $\hat{\theta}_A$ is large: sd$(\theta_A|y) \approx 10$.

0.10.2 3g.2 Cognitive Hierarchical Models: Individual Differences

The eight schools structure arises in cognitive science wherever the same task is administered to multiple participants. Each participant has an individual parameter θ_j (e.g., their drift rate, noise level, or prior probability), and these are drawn from a population distribution.

The cognitive science version: N participants complete a reaction time task. Model: $RT_{ij} \sim \text{LogNormal}(\mu + \theta_j, \sigma^2)$ where $\theta_j \sim \mathcal{N}(0, \tau^2)$ encodes individual deviation from the population mean μ. The hierarchical prior shrinks participants' individual estimates toward the group mean, yielding more stable individual estimates (less overfitting to individual data) while still capturing genuine individual differences.

The **credible interval for** τ tests the hypothesis of genuine individual differences: if τ's posterior is concentrated near zero, individual differences are negligible; if it is well away from zero, individual differences are real. This is the Bayesian alternative to the classical ANOVA F-test.

Non-centred parametrisation. The centred model $\theta_j \sim \mathcal{N}(\mu, \tau^2)$ creates a strong correlation between θ_j, μ, and τ in the posterior, making MCMC inefficient. The non-centred parametrisation $\theta_j = \mu + \tau\eta_j$ with $\eta_j \sim \mathcal{N}(0, 1)$ decouples the geometry: μ, τ, and η_j are approximately independent in the posterior when data are sparse (few observations per participant). This parametrisation, also called the **Matt trick** in Stan, is the most important practical improvement for hierarchical MCMC.

0.10.3 3g.3 Bayesian Meta-Analysis

Meta-analysis combines evidence from multiple studies to estimate an overall effect. The hierarchical model is identical to the eight schools model, with studies playing the role of schools. The **random-effects meta-analysis** (DerSimonian and Laird 1986) is the classical version; the Bayesian hierarchical version (Higgins and Spiegelhalter 2002) has several advantages:

1. The posterior for τ correctly accounts for the uncertainty in between-study variance, unlike the classical point estimate.

2. Weakly informative priors on τ prevent the unregularised estimate from being dominated by outlier studies.

3. Publication bias (the tendency to publish only statistically significant results) can be modelled explicitly by including a **selection model**: $p(\text{study published}|y_j, \sigma_j) = \Phi((y_j - c)/\sigma_j)$ for some threshold c, and correcting the posterior for this selection process.

The Bayesian meta-analysis posterior $p(\mu, \tau|y_{1:K})$ (for K studies) provides the full uncertainty over the population effect, enabling proper decision-making: given the posterior, the probability that the true effect exceeds a clinically meaningful threshold δ is $P(\mu > \delta|y_{1:K})$, computed directly from the posterior samples.

0.11 5i. Kernelised Stein Discrepancy and Goodness-of-Fit Testing

0.11.1 5i.1 Stein Operators and Discrepancy Measures

Given a distribution p with score function $\nabla_\theta \log p(\theta)$ and a test function g, the **Stein identity** states $\mathbb{E}_p[\mathcal{A}_p g(\theta)] = 0$ where $\mathcal{A}_p g = g(\theta)\nabla_\theta \log p(\theta) + \nabla_\theta g(\theta)$ is the Stein operator. The identity holds for any function g in Stein's class of p.

The **Kernelised Stein Discrepancy** (KSD; Liu et al., 2016; Chwialkowski et al., 2016):

$$S(q, p) = \sup_{\|g\|_{\mathcal{H}} \leq 1} \mathbb{E}_q[\mathcal{A}_p g(\theta)]^2,$$

measures how different q is from p using only the score function $\nabla_\theta \log p$ (not the normalising constant). The KSD has a closed-form expression in terms of the kernel k:

$$S(q, p) = \mathbb{E}_{\theta,\theta' \sim q}[u_p(\theta, \theta')],$$

where $u_p(\theta, \theta') = k_p(\theta, \theta') + \nabla_\theta k(\theta, \theta') \cdot s_p(\theta') + \nabla_{\theta'} k(\theta, \theta') \cdot s_p(\theta) + k(\theta, \theta') s_p(\theta)^\top s_p(\theta')$ and $s_p = \nabla_\theta \log p$.

Goodness-of-fit testing. For n samples from q, estimate $\hat{S}(q, p)$ and test $H_0 : q = p$. The test statistic $n\hat{S}$ converges to a weighted sum of chi-squared variables under H_0, enabling a valid test. KSD is particularly useful for testing unnormalised models (energy-based models, Boltzmann machines) where the normalising constant is unavailable.

0.12 14j. Counterfactual Reasoning in Cognitive Development

0.12.1 14j.1 Counterfactual Thinking in Children

Counterfactual reasoning — thinking about "what would have happened if ..." — develops gradually in children. Preschool children (3–4 years) can reason about immediate counterfactuals ("What would happen if the block weren't there?") but struggle with more distant counterfactual chains. School-age children (7–8 years) can reason about multi-step counterfactuals involving causal chains.

The Bayesian causal model predicts this developmental trajectory: Level 3 of Pearl's ladder of causation (counterfactuals) requires the full structural causal model (SCM), which requires accurate knowledge of the structural equations f_i. Young children's structural equations are poorly specified (they have causal intuitions but not precise functional relationships), making counterfactual inference unreliable. As children

accumulate causal knowledge, their structural equations become more precise, enabling more accurate counterfactual inference.

Formal counterfactual: For SCM $\mathcal{M} = (U, V, f)$, the counterfactual Y_x (the value Y would have had if X had been x) is computed by: (1) **Abduction**: infer U from observations **e** using $P(U|\mathbf{e})$. (2) **Action**: mutilate the SCM by replacing f_X with the constant x. (3) **Prediction**: compute Y in the mutilated model with U as inferred in step 1. The developmental trajectory maps to: (1) improving abductive inference (better inverse models), (2) understanding interventions (Theory of Mind milestone), (3) accurate forward prediction.

0.13 13h. Mutual Information Estimation in Practice

0.13.1 13h.1 The MINE Estimator

Computing mutual information $\mathrm{I}(X;Y)$ from samples is non-trivial: standard plug-in estimators (histogram-based) suffer from the curse of dimensionality in high dimensions. The **Mutual Information Neural Estimator** (MINE; Belghazi et al., 2018) uses the Donsker-Varadhan representation of KL divergence:

$$\mathrm{KL}(P\|Q) = \sup_{f:\Omega\to\mathbb{R}} \mathbb{E}_P[f] - \log \mathbb{E}_Q[e^f],$$

applied to $\mathrm{I}(X;Y) = \mathrm{KL}(P_{XY}\|P_X \otimes P_Y)$:

$$\mathrm{I}(X;Y) = \sup_{f_\theta} \mathbb{E}_{P_{XY}}[f_\theta(x,y)] - \log \mathbb{E}_{P_X\otimes P_Y}[e^{f_\theta(x,y)}],$$

where the supremum is over a neural network f_θ. MINE trains f_θ by gradient ascent on the MINE objective using minibatches from the joint distribution and the product of marginals (drawn by shuffling paired samples). The estimator converges to the true MI as the network class grows and the sample size increases.

Applications in neuroscience. MINE has been used to estimate: MI between neural activity in different brain areas, MI between stimulus features and neural responses (a measure of information coding), and MI between behaviour and neural states (measuring the information about upcoming decisions in pre-decision neural activity). All of these require high-dimensional MI estimation where classical methods fail.

0.14 20p. Numerical Cognition and the Approximate Number System

0.14.1 20p.1 Weber's Law and the ANS

The **Approximate Number System** (ANS) is the core non-symbolic numerical system shared by humans and many animals. It encodes numerosities (set sizes) as noisy Gaussian-like distributions on a logarithmically compressed mental number line:

$$p(\hat{n}|n) = \mathcal{N}(\hat{n}; \log n, w^2),$$

where w is the **Weber fraction** (approximately 0.15–0.25 in adult humans). The comparison of two numerosities n_1 and n_2 is decided by whether $|\log(n_1/n_2)| >$ threshold, giving a discrimination proba-

bility:

$$P(\text{correct}) = \Phi\left(\frac{|\log(n_1/n_2)|}{w\sqrt{2}}\right).$$

This is **Weber's Law**: discrimination accuracy depends only on the ratio n_1/n_2, not on the absolute values. The discriminability index $d' = |\log(n_1/n_2)|/(w\sqrt{2})$ decreases as the ratio approaches 1.

The Bayesian ANS model (Pica et al., 2004; Dehaene and Changeux 1993): the brain estimates numerosity by Bayesian inference combining the noisy perceptual representation $\hat{n}$ with a prior $p(n) \propto 1/n$ (scale-invariant, logarithmically flat). The posterior $p(n|\hat{n}) \propto \mathcal{N}(\hat{n}; \log n, w^2)/n$ gives the MAP estimate $n^* = \exp(\hat{n} - w^2)$, which underestimates large numerosities (the prior pulls the estimate toward smaller values). This explains the well-known **numerical subitizing effect**: small numerosities (1–4) are perceived exactly (because the perceptual noise w^2 is small relative to the spacing of log-integers), while larger numerosities are estimated approximately with Weber-law precision.

0.15 20o. Mathematical Models of Learning Strategies

0.15.1 20o.1 Exploration-Exploitation in Human Cognition

The **explore-exploit dilemma** in reinforcement learning has a direct cognitive analogue: should a person exploit current knowledge (choose the known-best option) or explore new options (gain information at the cost of immediate reward)? The optimal solution is the **Gittins index** (Gittins 1979): assign each option a dynamic allocation index $\nu_k(s_k)$ depending on its current state s_k, and always select the option with the highest index.

For the multi-armed bandit problem (K arms, each with unknown reward probability θ_k, $\theta_k \sim \text{Beta}(\alpha_k, \beta_k)$ prior), the Gittins index is:

$$\nu_k(\alpha_k, \beta_k) = \sup_{\tau \geq 0} \frac{\mathbb{E}\left[\sum_{t=0}^{\tau} \gamma^t r_t^k\right]}{\mathbb{E}\left[\sum_{t=0}^{\tau} \gamma^t\right]},$$

where the supremum is over stopping times τ and $r_t^k \sim \text{Bernoulli}(\theta_k)$. The Gittins index is the expected discounted reward-per-unit-time achievable from arm k before it is better to switch. The optimal policy selects $k^* = \arg\max_k \nu_k(\alpha_k, \beta_k)$ at each step, and updates $(\alpha_{k^*}, \beta_{k^*})$ based on the observed reward.

Human explore-exploit behaviour is often modelled by the **Thompson Sampling** policy: sample $\tilde{\theta}_k \sim \text{Beta}(\alpha_k, \beta_k)$ for each arm and select $k^* = \arg\max_k \tilde{\theta}_k$. Thompson sampling is asymptotically optimal (achieves the Lai-Robbins lower bound on cumulative regret) and is computationally trivial. Human bandit data fits Thompson sampling better than greedy or ϵ-greedy policies (Steyvers et al., 2009), providing evidence for Bayesian exploration in human cognition.

0.16 6e. Bayesian Neural Networks: Approximation Guarantees

0.16.1 6e.1 The PAC-Bayesian Framework for Neural Networks

The **PAC-Bayesian framework** (McAllester 1998; Catoni 2007) provides generalisation guarantees for Bayesian learning algorithms. For a prior π over hypotheses and any posterior ρ (possibly data-dependent),

the PAC-Bayes bound states: with probability $\geq 1 - \delta$ over the draw of n training examples,

$$\mathbb{E}_\rho[\text{Risk}(h)] \leq \mathbb{E}_\rho[\hat{\text{Risk}}(h)] + \sqrt{\frac{\text{KL}(\rho\|\pi) + \log(n/\delta)}{2n}},$$

where $\text{Risk}(h)$ is the true risk (expected loss) and $\hat{\text{Risk}}(h)$ is the empirical risk (training loss). The bound says: the expected true risk under ρ is bounded by the expected training loss plus a complexity term that grows with the KL divergence between the posterior and prior.

Consequences for BNNs. For a Bayesian neural network with Gaussian prior $\pi = \mathcal{N}(0, \sigma_0^2 I)$ and posterior $\rho = \mathcal{N}(\mathbf{w}_{\text{MAP}}, \Sigma)$:

$$\text{KL}(\rho\|\pi) = \frac{1}{2\sigma_0^2}(\|\mathbf{w}_{\text{MAP}}\|^2 + \text{tr}(\Sigma)) - \frac{p}{2} - \frac{1}{2}\log\det(\sigma_0^{-2}\Sigma) - \frac{p}{2}.$$

This is minimised by finding the posterior ρ that best balances low training loss (small $\mathbb{E}_\rho[\hat{\text{Risk}}]$) and low complexity (small $\text{KL}(\rho\|\pi)$). The PAC-Bayes bound provides a principled objective for training BNNs: minimise the right-hand side to obtain the tightest generalisation guarantee.

0.17 9l. The Free Energy Principle: A Critical Assessment

0.17.1 9l.1 Strengths

The Free-Energy Principle has several genuine mathematical strengths:

Unification. The FEP provides a single variational principle that unifies perception, action, learning, attention, development, and evolution under one mathematical umbrella. This unification is genuinely unprecedented: no previous theory of brain function connects all these phenomena in a single principled framework.

Derivability. Many known neuroscientific phenomena — Hebbian plasticity (STDP), opponent processing in sensory systems, attention as precision weighting, dopamine as prediction error, the hierarchical organisation of cortex — can be derived as consequences of free-energy minimisation. This derivability (rather than mere consistency) is a strong theoretical virtue.

Quantitative predictions. The FEP generates quantitative, testable predictions (Section 9f, 9i): specific reaction time distributions, specific patterns of neural activity in predictive coding, specific effects of dopamine depletion on precision weighting.

0.17.2 9l.2 Criticisms and Limitations

Criticism 1: Trivial claim. Some critics (Colombo and Series 2012) argue that the FEP is trivially true for any system at NESS: any probability distribution can be expressed as a free energy, so the claim that living systems "minimise free energy" is an analytical truth, not an empirical prediction. The FEP becomes empirically interesting only when the specific form of the generative model is specified; the general principle is unfalsifiable.

Criticism 2: The non-uniqueness problem. The decomposition of a system's dynamics into a "gradient flow" (free-energy descent) and a "solenoidal" (divergence-free) component is not unique: different

choices of the diffusion tensor Q give different decompositions. The claim that the system is "minimising free energy" picks out one particular decomposition, but other decompositions would give different descriptions of the same dynamics. The FEP description is a mathematical convenience, not a causal claim.

Criticism 3: Computability. For large-scale neural systems, the free energy $\mathcal{F}$ is not computable in principle (it requires a fully specified generative model of the entire universe), and not tractable in practice. The predictive coding implementation is an approximation; the quality of the approximation is model-dependent.

Response. Friston (2019) argues that these criticisms confuse the mathematical principle with its implementation: the FEP is a formal description of self-organisation at NESS, not an algorithmic prescription. The non-uniqueness of the decomposition is addressed by selecting the decomposition that maps to the known biological architecture (the Markov blanket decomposition). The computability objection is addressed by noting that all scientific models are approximations; the FEP's approximations are no worse than those of other computational theories.

0.18 3o. Imprecise Probability and Robust Bayesian Analysis

0.18.1 3o.1 Credal Sets and Sensitivity Analysis

Imprecise probability (Walley 1991; de Cooman and Troffaes 2014) generalises probability theory by representing beliefs as a set of compatible probability measures (a **credal set**) rather than a single distribution. This is appropriate when: (1) the prior is not precisely known, (2) expert elicitation produces ranges rather than point probabilities, or (3) the data are insufficient to determine the posterior precisely.

A credal set $\mathcal{P} = \{P_\theta : \theta \in \Theta\}$ is a set of probability measures parametrised by a hyperparameter $\theta \in \Theta$. The **lower probability** $\underline{P}(A) = \inf_\theta P_\theta(A)$ and **upper probability** $\overline{P}(A) = \sup_\theta P_\theta(A)$ are the most pessimistic and optimistic assessments of event A. The interval $[\underline{P}(A), \overline{P}(A)]$ represents the agent's imprecise belief about A.

Sensitivity analysis. In Bayesian analysis, sensitivity analysis varies the prior over a credal set and examines how much the posterior varies: $[\underline{P}(\theta|D), \overline{P}(\theta|D)] = [\inf_\theta P_\theta(\theta|D), \sup_\theta P_\theta(\theta|D)]$. Robust conclusions are those where the posterior lower and upper probabilities agree; fragile conclusions are those where they disagree.

For cognitive science, imprecise probability models the bounded rationality of real agents: a rational agent with limited information may justifiably represent their beliefs as a credal set rather than a single distribution. Decision-making under imprecise probability uses the **maximin expected utility**: $a^* = \arg\max_a \inf_{P \in \mathcal{P}} \mathbb{E}_P[U(a)]$ (the action that maximises the worst-case expected utility).

0.19 9k. The Mathematics of Intentions and Planning

0.19.1 9k.1 Hierarchical Planning and Goal Decomposition

Hierarchical task networks (HTN; Erol et al., 1994) model planning as the decomposition of high-level goals into primitive tasks. An HTN has:

- **Primitive tasks**: executable actions with preconditions and effects.
- **Compound tasks**: abstract actions that decompose into other tasks.
- **Decomposition methods**: if compound task T holds in context C, replace T with subtask network N.

HTN planning is **undecidable** in general (it is as powerful as Turing computation), but becomes tractable with restrictions. **Total order** HTN planning (where subtasks are ordered) is NP-hard; **partial order** planning (where subtasks may be unordered) is PSPACE-hard.

Bayesian HTN. A probabilistic HTN places distributions over decomposition methods: $P(N|T, C)$ is the probability of decomposing task T by method N in context C. The most probable plan (MPP) is the plan with highest probability under the hierarchical generative model:

$$P(\text{plan}) = \prod_{T,N \text{ used}} P(N|T, C) \prod_{a \text{ primitive}} P(\text{success}(a)|s).$$

Bayesian HTN planning is used in cognitive architectures (ACT-R, SOAR) to model intentional action: the agent selects decompositions with probability proportional to their expected utility.

0.20 13g. Information in Neural Circuits: A Synthesis

0.20.1 13g.1 From Shannon to Neurons: Bridging Theory and Biology

Shannon's information theory (Section VI) provides the mathematical language for describing neural coding, but the connection between theory and biology requires several translation steps:

Step 1: Defining the channel. The neural "channel" maps a stimulus $s \in \mathcal{S}$ to a spike train $\mathbf{r}$. The mutual information $\mathrm{I}(S; \mathbf{r}) = H(\mathbf{r}) - H(\mathbf{r}|S)$ measures the average information transmitted per trial. Computing $H(\mathbf{r})$ requires knowing the full distribution over spike trains, which is difficult experimentally; it is often approximated by the linear Fisher information bound $\mathrm{I}(S; \mathbf{r}) \leq \frac{1}{2} \log \det(I + \mathcal{I}(S))$ (using the data processing inequality and the Gaussian channel upper bound).

Step 2: Rate vs timing. For rate codes, $H(\mathbf{r}) \approx H(\bar{r}) + H(\text{variability}|\bar{r})$. The first term is the information in the rate; the second is the information in the variability around the rate (which may include temporal information). For a Poisson neuron, variability is entirely noise ($H(\text{variability}|\bar{r}) \leq 0$ contribution to MI), but for non-Poisson neurons, variability carries information.

Step 3: Population codes. For a population of N neurons, the Fisher information matrix $\mathcal{I}_{\text{pop}}$ (Section 12c) bounds the encoding precision. The information-geometric Section of the posterior $p(s|\mathbf{r})$ scales as $\det(\mathcal{I}_{\text{pop}})^{-1/2}$: larger Fisher information → smaller posterior Section → more precise decoding.

0.21 26j. Open Problems in Mathematical Cognitive Science

0.21.1 26j.1 Unsolved Mathematical Problems

We collect the most important open mathematical problems that will shape the next decade of mathematical cognitive science:

Problem 1: The complexity of IIT. Prove or disprove: there exists a polynomial-time approximation scheme (PTAS) for computing Φ with approximation ratio $(1+\epsilon)$ for any $\epsilon > 0$. The existence of a PTAS would make IIT computationally practical; its impossibility (under standard complexity assumptions) would demonstrate the fundamental intractability of consciousness measurement.

Problem 2: Identifiability of Active Inference agents. Given a finite sequence of observations and actions from an Active Inference agent with unknown generative model $p(\tilde{o}, \tilde{s})$ and preference $\tilde{P}(\tilde{o})$, characterise the set of generative models and preferences that are consistent with the observed behaviour. What additional interventions (e.g., pharmacological modulation of precision) suffice for identifiability?

Problem 3: Universal approximation with positional bounds. The Universal Approximation Theorem guarantees that neural networks can approximate any continuous function on a compact set, but provides no bound on the required network size. Prove or disprove: for any continuous function $f : [0,1]^d \to \mathbb{R}$ with Lipschitz constant L, a ReLU network with $O(\epsilon^{-d} L^d)$ neurons achieves $\sup |f - f_{\text{net}}| \leq \epsilon$.

Problem 4: Sample complexity of causal discovery. Determine the tight sample complexity for recovering the Markov equivalence class of a causal graph $\mathcal{G}$ with n nodes, maximum degree k, and bounded path lengths. Known upper bounds from the PC algorithm are $O(\text{poly}(n,k) \cdot \log(1/\delta))$; matching lower bounds are not fully established.

Problem 5: The de Finetti theorem for exchangeable belief functions. Extend the de Finetti representation theorem (Section 3b) to **imprecise probability** (lower and upper probabilities, Dempster-Shafer belief functions). Under what conditions does an exchangeable sequence of belief functions admit a de Finetti-type representation as a mixture of i.i.d. sequences of belief functions?

0.22 4d. Bayesian Nonparametric Survival Analysis and Cognitive Aging

0.22.1 4d.1 Survival Analysis and the Hazard Function

Survival analysis studies the time until an event occurs: e.g., time to memory failure, reaction time (time to response), or age at cognitive decline. The key quantity is the **hazard function** $h(t) = \lim_{\Delta t \to 0} P(t \leq T < t + \Delta t | T \geq t)/\Delta t$: the instantaneous rate of the event given survival to time t. The survival function $S(t) = P(T > t) = \exp(-\int_0^t h(s)\, ds)$.

Common parametric models:

- **Exponential**: $h(t) = \lambda$ (constant hazard). $S(t) = e^{-\lambda t}$. Models memoryless events.
- **Weibull**: $h(t) = (\nu/\lambda)(t/\lambda)^{\nu-1}$. $S(t) = \exp(-(t/\lambda)^\nu)$. Increasing ($\nu > 1$), decreasing ($\nu < 1$), or constant ($\nu = 1$) hazard.
- **Log-normal**: $\log T \sim \mathcal{N}(\mu, \sigma^2)$. Models events with log-normal latency distributions.

Bayesian inference for survival models: place priors over the hazard parameters and compute the posterior given observed event times (and censored observations — times at which the event had not yet occurred). The posterior predictive survival function $P(T > t|\text{data}) = \int S(t|\lambda) p(\lambda|\text{data})\, d\lambda$ accounts for parameter uncertainty.

0.22.2 4d.2 The Cox Proportional Hazards Model

The **Cox model** (Cox 1972) separates the baseline hazard $h_0(t)$ from the effect of covariates $\mathbf{x}$:

$$h(t|\mathbf{x}) = h_0(t)\exp(\boldsymbol{\beta}^\top \mathbf{x}).$$

The partial likelihood (eliminating the baseline hazard) for ordered event times $t_{(1)} < \cdots < t_{(n)}$:

$$L(\boldsymbol{\beta}) = \prod_{i=1}^{n} \frac{\exp(\boldsymbol{\beta}^\top \mathbf{x}_i)}{\sum_{j \in R(t_{(i)})} \exp(\boldsymbol{\beta}^\top \mathbf{x}_j)},$$

where $R(t)$ is the risk set at time t (all subjects who have not yet had the event or been censored). Maximising L gives the Cox MLE, which is efficient for estimating $\boldsymbol{\beta}$ without specifying h_0.

Bayesian Cox model: place a Gaussian prior $\boldsymbol{\beta} \sim \mathcal{N}(0, \sigma_\beta^2 I)$ and compute the posterior by MCMC. The baseline hazard h_0 can be modelled nonparametrically using a **Gamma process** prior (Kalbfleisch 1978): $H_0(t) = \int_0^t h_0(s)\,ds \sim$ Gamma process$(\alpha H_0^*(t), \alpha)$ where H_0^* is a baseline cumulative hazard and α is a concentration parameter. The Gamma process prior gives conjugate updating with the Cox likelihood.

0.23 18h. Causality and Interventional Probabilistic Programming

0.23.1 18h.1 Causal Probabilistic Programs

A **causal probabilistic program** extends a probabilistic program with explicit $\texttt{do}(X = x)$ operators that implement Pearl's do-operator. In a causal program, each random variable is generated by a structural equation $X_i = f_i(\text{PA}_i, U_i)$ where PA_i are the parents and U_i is exogenous noise. The do-operator mutilates the program: $\texttt{do}(X_j = x)$ replaces the structural equation for X_j with the constant function $X_j = x$, removing all edges into X_j.

A causal probabilistic program can answer all three levels of Pearl's ladder of causation:

- **Level 1 (observational)**: $P(Y = y|X = x)$ — run the program and condition.
- **Level 2 (interventional)**: $P(Y = y|\texttt{do}(X = x))$ — mutilate the program and run.
- **Level 3 (counterfactual)**: $P(Y_x = y|X = x', Y = y')$ — abduction (infer U from observations), mutilate, and predict.

This is the program-level implementation of the do-calculus: the structural equations provide the causal model, and the program semantics provides the probabilistic inference.

0.24 22i. Mathematical Linguistics: Formal Language Theory

0.24.1 22i.1 The Chomsky Hierarchy

Chomsky's hierarchy (Chomsky 1956) classifies formal grammars by their generative power:

Type	Grammar	Automaton
0	Unrestricted	Turing machine
1	Context-sensitive	Linear-bounded automaton
2	Context-free	Pushdown automaton
3	Regular	Finite automaton

Each type is strictly more powerful than the next: Type 3 $\subsetneq$ Type 2 $\subsetneq$ Type 1 $\subsetneq$ Type 0. Natural language syntax appears to require at least mildly context-sensitive grammars: English generates cross-serial dependencies (as in Dutch) that are not context-free, but natural language also appears to be more restricted than full context-sensitivity.

The Pumping Lemma for context-free languages: if L is a CFL and $|s| \geq p$ (the pumping length), then $s = uvwxy$ with $|vx| \geq 1$, $|vwx| \leq p$, and $uv^n wx^n y \in L$ for all $n \geq 0$. Languages that violate this lemma are not CFLs: $\{a^n b^n c^n\}$ violates the pumping lemma (cannot simultaneously pump as and cs without also pumping bs), placing it outside the CFL class. This is a mathematical demonstration that some natural language constructions (like the Swiss-German verb-raising pattern) require grammars beyond CFG.

0.24.2 22i.2 Probabilistic Context-Free Grammars

A **Probabilistic Context-Free Grammar** (PCFG) extends a CFG with probabilities over production rules. For each nonterminal A, the productions $A \to \alpha_1, A \to \alpha_2, \ldots$ have associated probabilities $p_1, p_2, \ldots$ with $\sum_i p_i = 1$. The probability of a parse tree T is the product of the probabilities of all rules used:

$$P(T) = \prod_{A \to \alpha \in T} P(A \to \alpha).$$

The **Inside-Outside algorithm** computes the expected sufficient statistics $\mathbb{E}[c(A \to \alpha | \mathbf{w})]$ (the expected number of times rule $A \to \alpha$ is used to parse sentence $\mathbf{w}$) in $O(n^3 |G|)$ time. These expected counts can be used for maximum likelihood estimation of rule probabilities from an unannotated corpus (the EM algorithm for PCFGs):

$$P^{\text{new}}(A \to \alpha) = \frac{\sum_{\mathbf{w}} \mathbb{E}[c(A \to \alpha | \mathbf{w})]}{\sum_{\mathbf{w}} \sum_{\alpha} \mathbb{E}[c(A \to \alpha | \mathbf{w})]}.$$

Bayesian estimation of PCFG parameters uses a Dirichlet prior $P(A \to \alpha) \sim \text{Dir}(\alpha_0)$ for each nonterminal A. The posterior updates the Dirichlet hyperparameters by adding the expected rule counts; the posterior predictive distribution over new sentences averages over the uncertainty in rule probabilities.

0.25 3n. The Bernstein-von Mises Theorem

0.25.1 3n.1 Bayesian Asymptotics

As the number of observations grows, the posterior distribution concentrates around the true parameter and becomes increasingly Gaussian, regardless of the prior. This is the content of the **Bernstein-von Mises theorem**:

Theorem 0.25.1 (Bernstein-von Mises)**.** Let $X_1, \ldots, X_n \overset{\text{i.i.d.}}{\sim} p(\cdot|\theta_0)$ and let $\pi(\theta)$ be a prior with $\pi(\theta_0) >$

0. Under regularity conditions (identifiability, differentiability of the log-likelihood, Fisher information finite and positive at θ_0), as $n \to \infty$:

$$\sup_A \left| P(\sqrt{n}(\theta - \hat{\theta}_n) \in A \mid X^n) - \mathcal{N}(A; 0, \mathcal{I}(\theta_0)^{-1}) \right| \xrightarrow{P_{\theta_0}} 0,$$

where $\hat{\theta}_n$ is the MLE and $\mathcal{I}(\theta_0)$ is the Fisher information at the true parameter.

The theorem says: the posterior concentrates around the MLE at rate $1/\sqrt{n}$ and is asymptotically normal with covariance $\mathcal{I}^{-1}/n$. In the limit, Bayesian and frequentist inference agree: the posterior credible interval is asymptotically the same as the frequentist confidence interval. The prior washes out as $n \to \infty$.

Implications for cognitive science. For a Bayesian learner with many observations, the choice of prior matters less and less: eventually, the data swamp the prior and the posterior is determined by the likelihood. This explains the empirical convergence of expert judgments: experienced domain experts (who have seen many relevant examples) agree more with each other and more with objective frequency estimates than novices (who are more influenced by their subjective priors).

0.25.2 3n.2 Non-parametric Bernstein-von Mises

For non-parametric models (where the parameter is a function, not a finite-dimensional vector), the BvM theorem may fail or hold in a weaker form. The **Diaconis-Freedman phenomenon** (1986) shows that the posterior can be inconsistent (fail to concentrate at the truth) for some non-parametric models and priors, even with Gaussian process priors on smooth function spaces.

For GP regression with n observations of a smooth function, the posterior concentrates at rate $n^{-s/(2s+1)}$ for a function in a Sobolev space of order s — faster than the parametric $n^{-1/2}$ for very smooth functions. The covariance of the posterior does not converge to a fixed Gaussian; instead, it shrinks to zero at the same rate as the posterior mean error. This is the non-parametric analogue of the BvM theorem.

0.26 7k. Sparse Gaussian Processes: Inducing Point Methods

0.26.1 7k.1 The Inducing Point Framework

Exact GP inference costs $O(n^3)$ in time and $O(n^2)$ in memory. For large datasets ($n > 10^4$), sparse approximations are essential. The **inducing point framework** introduces $m \ll n$ inducing inputs $Z = (z_1, \ldots, z_m)$ with associated inducing outputs $\mathbf{u} = (f(z_1), \ldots, f(z_m))$. All approximations approximate the conditional $p(f|X)$ by conditioning on $\mathbf{u}$ instead of on the full function.

The exact conditional is $p(\mathbf{f}|X) = \int p(\mathbf{f}|\mathbf{u})p(\mathbf{u})\, d\mathbf{u}$ (integrating out $\mathbf{u}$ recovers the full GP). Approximations restrict the dependence:

$$q(\mathbf{f}|\mathbf{u}) \approx p(\mathbf{f}|\mathbf{u}), \quad q(\mathbf{u}) = \mathcal{N}(\mathbf{u}; \mathbf{m}, S).$$

The variational sparse GP (Titsias 2009) optimises $(\mathbf{m}, S, Z)$ by maximising a lower bound on $\log p(\mathbf{y}|X)$:

$$\mathcal{L} = \sum_i \mathbb{E}_{q(f_i)}[\log p(y_i|f_i)] - \mathrm{KL}(q(\mathbf{u}) \| p(\mathbf{u})).$$

The optimal $q(\mathbf{u})$ has $\mathbf{m} = K_{ZZ}^{-1}K_{ZX}(K_{XX|Z}+\sigma^2 I)^{-1}\mathbf{y}$ and $S = K_{ZZ}^{-1}K_{ZZ}^{-1}(K_{ZZ}+K_{ZX}\Lambda K_{XZ})^{-1}K_{ZZ}$ (where $\Lambda = \text{diag}(1/\sigma_i^2)$). The approximation costs $O(nm^2 + m^3)$ per gradient step.

0.27 11k. Consciousness and Integrated Information: Experimental Predictions

0.27.1 11k.1 Testable Predictions of IIT

IIT makes several concrete predictions that distinguish it from other theories of consciousness:

Prediction 1: The cerebellum is not conscious. The cerebellum has $\sim$70 billion neurons (more than the rest of the brain combined) but is almost entirely feedforward (cerebellar granule cells receive input and project to Purkinje cells with minimal recurrence). IIT predicts $\Phi_{\text{cerebellum}} \approx 0$ (feedforward networks have no integrated information, since any bipartition into input and output units produces MIP $\approx$ full system). Consistent with: cerebellar lesions cause motor coordination deficits but not loss of consciousness.

Prediction 2: Spinal cord reflexes are unconscious. Spinal reflex arcs (e.g., the knee-jerk reflex) are local, feedforward circuits without recurrent integration. IIT predicts $\Phi \approx 0$ for the reflex circuit in isolation. Consistent with: spinal cord reflexes occur in paraplegic patients (who have lost conscious sensation below the lesion) and in anaesthetised subjects.

Prediction 3: Deep sleep = low Φ. During NREM slow-wave sleep, cortical dynamics break down into local, bistable up-down states with little long-range integration. IIT predicts $\Phi_{\text{NREM}} \ll \Phi_{\text{waking}}$. Consistent with: PCI measurements (Section 11e) show reduced complexity during deep sleep; TMS-evoked cortical responses lose long-range propagation.

Prediction 4: Split-brain patients have two conscious systems. Sectioning the corpus callosum (the main interhemispheric connection) should split the cortex into two maximally integrated subsystems, each with its own Φ^*. IIT predicts two separate conscious experiences. Consistent with: split-brain patients show independent left- and right-hemisphere responses in divided visual field experiments, suggesting two independent streams of consciousness.

0.28 22j. The Logical Structure of Mathematical Proof

0.28.1 22j.1 Proof Theory and the Curry-Howard Isomorphism

The **sequent calculus** (Gentzen 1935) is the foundational proof system for classical and intuitionistic logic. A **sequent** $\Gamma \vdash \Delta$ asserts that some formula in Δ is provable from the assumptions in Γ. The calculus has structural rules (weakening, contraction, cut) and logical rules (introduction and elimination for each connective).

The **Cut Elimination Theorem** (Gentzen's *Hauptsatz*) states that every proof in the sequent calculus can be transformed into a cut-free proof. Cut corresponds to the use of a lemma: proving B using A as an intermediate step. Cut-elimination shows that lemmas are never necessary: any valid proof can be written without them. Computationally (via Curry-Howard), cut-elimination is function composition reduction: replacing $f(g(x))$ by the direct computation.

The theorem has profound implications for cognitive science: it suggests that chains of reasoning (using

intermediate conclusions as stepping stones) can always be replaced by a single direct argument. However, the cut-free proof may be exponentially longer than the proof with cuts: lemmas provide exponential compression. Human mathematical reasoning typically uses lemmas (intermediate results), suggesting that cognitive efficiency favours cut-enabled proofs over cut-free ones — a direct consequence of the fact that cut-free proofs can be exponentially longer.

0.29 11j. The Exclusion Postulate: Mathematical Consequences

0.29.1 11j.1 The Exclusion Problem

IIT's exclusion postulate says that of all possible systems that could be defined over a physical substrate, only the one with maximum Φ is the actual conscious system. This prevents the "grain problem" (what determines the spatial and temporal grain of the system that is conscious?) by selecting the unique maximally integrated partition.

Mathematically, the exclusion postulate is equivalent to a **maximum selection** over a discrete, exponentially large search space: for n physical elements, there are $2^n - 1$ candidate systems (non-empty subsets), each with a Φ value. The exclusion postulate selects $S^* = \arg\max_S \Phi(S)$.

The problem: this maximum may not be unique. Two systems with equal Φ would violate the exclusion postulate's requirement of a single conscious system. IIT 4.0 resolves this by introducing a lexicographic tie-breaking rule (favour smaller systems, then favour systems with higher average ϕ per mechanism). The mathematical uniqueness of the maximum under this rule is not guaranteed for all network architectures.

0.30 20n. Cognitive Maps and the Hippocampus

0.30.1 20n.1 O'Keefe and Nadel's Cognitive Map Theory

O'Keefe and Nadel (1978) proposed that the hippocampus implements a **cognitive map**: a spatial representation of the environment that supports flexible, goal-directed navigation. The cognitive map is a geometric structure (a metric space $\mathcal{M}$ with distance function d) in which locations are represented by patterns of place cell activity.

The **vector space structure** of the hippocampal representation (Bush et al., 2015; Banino et al., 2018) has been formalised as: any goal-directed trajectory from current position $\mathbf{r}$ to target $\mathbf{r}^*$ can be computed as the vector difference $\Delta\mathbf{r} = \mathbf{r}^* - \mathbf{r}$ in the cognitive map's vector space, and this vector can be decoded into a motor command by the entorhinal-hippocampal circuit. Navigation is vector arithmetic in a spatial coordinate system.

0.30.2 20n.2 Path Integration and Dead Reckoning

Path integration (dead reckoning) is the computation of current position from a known starting position and a record of self-motion (speed, direction, duration). Formally: $\mathbf{r}(t) = \mathbf{r}(0) + \int_0^t \mathbf{v}(s)\, ds$ where $\mathbf{v}(s)$ is the velocity at time s.

In the brain, path integration is implemented by the entorhinal grid cells (Section 13d): the periodic firing pattern of grid cells is updated by the animal's velocity input, maintaining an estimate of position even in the dark (without sensory landmarks). The attractor network model (McNaughton et al., 2006): the grid

cell population is a continuous attractor network in which the bump of activity is shifted by the velocity input. The mathematical dynamics:

$$\dot{\mathbf{r}} = \mathbf{v}(t), \quad \text{(path integration)}$$

$$r_i(\mathbf{r}) = f(\mathbf{g}_i \cdot \mathbf{r} + \phi_i), \quad \text{(grid cell response)}$$

where $\mathbf{g}_i$ is the grid vector, ϕ_i is the phase offset, and f is the rectifying activation function. The path integration equation drives the phase of the grid cell activity to track the animal's position.

0.31 26i. Ecological Psychology and the Mathematics of Affordances

0.31.1 26i.1 Affordances as Information

Gibson's (1979) **affordance** theory proposes that the environment offers action possibilities to the animal — affordances — and that perception is the direct detection of these affordances, not the inference of them from retinal images. The animal detects "can I fit through that gap?" not "what is the shape of that gap?" followed by "given my body size, can I fit through?"

The **optical invariants** that specify affordances are mathematical properties of the optic array (the structured light reaching the eye) that are invariant under observer movement and transformation. The key invariant for gap passability is the ratio of shoulder width to gap width, detectable from the optic array without computing absolute distance or size.

Formally, for a gap of width W at distance D, the **visual angle** subtended is $\alpha = 2\arctan(W/2D) \approx W/D$ (for small angles). The animal's shoulder width at the same distance subtends angle $\beta = S/D$ (where S is shoulder width). The ratio $\alpha/\beta = W/S$ is invariant under changes in D: it specifies the passability affordance independently of distance. The animal need not estimate W, D, or S separately; it only needs to detect the invariant ratio α/β, which is directly available in the optic array.

0.31.2 26i.2 Motor System and Affordance Perception

Embodied affordance detection requires that the perceptual system be calibrated to the animal's action capabilities. An animal that has grown (its action capabilities changed) must recalibrate its affordance detection. Neurophysiologically, the motor system (premotor cortex, supplementary motor area) is active during affordance perception, even without overt action — consistent with the **simulation** view that affordance detection involves implicit motor simulation.

The information-theoretic formalisation (Cisek and Kalaska 2010): affordances are encoded as **potential motor programs** in premotor cortex. The posterior probability of affordance a given optic array $\mathbf{x}$ is $P(a|\mathbf{x}) \propto P(\mathbf{x}|a)P(a)$ where $P(a)$ is the prior probability of affordance a (based on the current context and task) and $P(\mathbf{x}|a)$ is the likelihood of observing optic array $\mathbf{x}$ given affordance a. Action selection is then $a^* = \arg\max_a P(a|\mathbf{x})U(a)$ where $U(a)$ is the utility of affordance a for the current goal.

0.32 3m. Sequential Monte Carlo and Particle Filters

0.32.1 3m.1 The Particle Filter for Online Bayesian Filtering

The **particle filter** (also: Sequential Importance Resampling, SIR; Gordon et al., 1993) is an approximate Bayesian filter for nonlinear, non-Gaussian state-space models. It represents the posterior $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ by a set of N weighted particles $\{(\mathbf{x}_t^{(i)}, w_t^{(i)})\}_{i=1}^N$:

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}) \approx \sum_{i=1}^{N} w_t^{(i)} \delta(\mathbf{x}_t - \mathbf{x}_t^{(i)}).$$

The SIR algorithm at each time step:

1. **Propagate**: draw $\mathbf{x}_{t+1}^{(i)} \sim p(\mathbf{x}_{t+1}|\mathbf{x}_t^{(i)})$ from the transition model.
2. **Weight**: set $w_{t+1}^{(i)} \propto w_t^{(i)} \cdot p(\mathbf{y}_{t+1}|\mathbf{x}_{t+1}^{(i)})$.
3. **Normalise**: $\tilde{w}^{(i)} = w^{(i)} / \sum_j w^{(j)}$.
4. **Resample**: draw N particles with replacement from $\{(\mathbf{x}^{(i)}, \tilde{w}^{(i)})\}$; reset weights to $1/N$.

The particle filter is consistent: as $N \to \infty$, the approximation converges to the true posterior. For finite N, the variance of estimates scales as $O(1/N)$ by the central limit theorem. The effective sample size $\text{ESS} = (\sum_i \tilde{w}^{(i)})^2 / \sum_i (\tilde{w}^{(i)})^2$ tracks weight degeneracy; resampling is triggered when ESS falls below a threshold (typically $N/2$).

Optimal proposal. The SIR filter uses the transition model as the proposal. The optimal proposal minimises the variance of the importance weights:

$$q^*(\mathbf{x}_{t+1}|\mathbf{x}_t^{(i)}, \mathbf{y}_{t+1}) = p(\mathbf{x}_{t+1}|\mathbf{x}_t^{(i)}, \mathbf{y}_{t+1}) \propto p(\mathbf{y}_{t+1}|\mathbf{x}_{t+1}) p(\mathbf{x}_{t+1}|\mathbf{x}_t^{(i)}).$$

When the optimal proposal is available (e.g., for conjugate models), using it reduces the variance dramatically and allows N to be much smaller.

0.32.2 3m.2 Rao-Blackwellised Particle Filters

When the state space has both discrete (nonlinear) and continuous (linear-Gaussian) components, the **Rao-Blackwellised particle filter** (RBPF; Doucet et al., 2000) marginalises out the linear-Gaussian components analytically (using Kalman filtering) and only uses particles for the nonlinear components.

For a state $\mathbf{x} = (\mathbf{z}, \boldsymbol{\theta})$ where $\mathbf{z}$ is the nonlinear component and $\boldsymbol{\theta}$ is the linear-Gaussian component, each particle i carries a Kalman filter state $(\hat{\boldsymbol{\theta}}^{(i)}, P^{(i)})$ for the linear component. The particle weight is the marginal likelihood of the observations under the Kalman filter:

$$w_t^{(i)} \propto p(\mathbf{y}_{1:t}|\mathbf{z}_{1:t}^{(i)}) = \prod_s \mathcal{N}(\mathbf{y}_s; H\hat{\boldsymbol{\theta}}_{s|s-1}^{(i)}, S_s^{(i)}),$$

where $S_s^{(i)} = H P_{s|s-1}^{(i)} H^\top + R$ is the innovation covariance. RBPF achieves the statistical efficiency of $N \cdot d_\theta$ particles with only N particles, where d_θ is the dimension of the linear-Gaussian component.

Cognitive application. The RBPF is a natural model of **cognitive tracking**: tracking an object's continuous trajectory (linear-Gaussian position and velocity) while inferring its discrete category (which object is it?). The particle filter handles category uncertainty; the Kalman filter handles continuous trajectory estimation. The combined system provides Bayesian tracking with automatic category learning.

0.33 3l. Exchangeability in Practice: Bayesian Nonparametrics

0.33.1 3l.1 Exchangeable Sequences and the Pólya Urn

The **Pólya urn model** is the simplest exchangeable sequence. Start with an urn containing α red and β blue balls. At each step, draw a ball, observe its colour, replace it, and add one ball of the same colour. The sequence of colours $X_1, X_2, \ldots$ is exchangeable.

The joint probability of the first n draws is:

$$P(X_1 = x_1, \ldots, X_n = x_n) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \cdot \frac{\Gamma(\alpha+k)\Gamma(\beta+n-k)}{\Gamma(\alpha+\beta+n)},$$

where $k = \sum_i x_i$ (number of reds). By de Finetti's theorem, this is a mixture of i.i.d. Bernoulli sequences: the directing measure μ is the Beta(α, β) distribution. The Pólya urn is thus the data-generating process of the Beta-Bernoulli conjugate model (Section 3d): the predictive probability after n draws is:

$$P(X_{n+1} = 1 | X_{1:n}) = \frac{\alpha + k}{\alpha + \beta + n},$$

which is the posterior mean of a Beta$(\alpha+k, \beta+n-k)$ distribution. This is the Laplace rule of succession (for $\alpha = \beta = 1$) or more generally the **Bayes-Laplace estimate** of probability.

0.33.2 3l.2 Exchangeable Partitions: the Ewens Sampling Formula

When the de Finetti representation is a Dirichlet Process (Section 6b), the joint distribution over cluster assignments $z_1, \ldots, z_n$ (which cluster each observation belongs to) follows the **Ewens Sampling Formula**:

$$P(z_1, \ldots, z_n) = \frac{\alpha^{K_n}}{\alpha^{(n)}} \prod_{k=1}^{K_n} (n_k - 1)!,$$

where K_n is the number of distinct clusters, n_k is the size of cluster k, and $\alpha^{(n)} = \alpha(\alpha+1)\cdots(\alpha+n-1)$ is the rising factorial. This distribution is **exchangeable as a partition**: the probability depends only on the partition sizes $n_1, \ldots, n_{K_n}$, not on which elements are in which cluster.

The expected number of clusters is $\mathbb{E}[K_n] = \alpha \sum_{k=1}^{n} 1/(alpha + k - 1) \approx \alpha \log(1 + n/\alpha)$: logarithmic growth in n. This models the cognitive observation that the number of perceived categories grows sublinearly with the number of stimuli: after seeing many examples, the learner creates fewer new categories per additional example.

0.34 3k. A Complete Bayesian Cognitive Model: The Rational Learner

0.34.1 3k.1 Model Architecture

We develop a complete Bayesian model of **word learning**: how a child learns the meaning of a new word from a few examples. This **rational model of word learning** (Xu and Tenenbaum 2007) is the clearest demonstration of Bayesian inference in cognitive development.

The task. A child is shown n examples of a new word (e.g., a teacher points to three different Dalmatians and says "dax"). What objects would the child extend "dax" to? Does "dax" mean dog, Dalmatian, or something else?

Hypothesis space. The learner considers a hierarchy of hypotheses H: from the most specific ($H_1 =$ only this exact dog) to the most general ($H_5 =$ all animals), with intermediate levels ($H_2 =$ Dalmatians, $H_3 =$ dogs, $H_4 =$ mammals). Each hypothesis defines a **concept** (a set of objects).

Likelihood: the Principle of Suspicious Coincidence. If the teacher is sampling uniformly at random from the extension of the word, the likelihood of observing n examples from H_k (with extension of size $|H_k|$) is:

$$P(\mathbf{x}_{1:n}|H_k) = \left(\frac{1}{|H_k|}\right)^n.$$

For $n = 3$ examples all showing Dalmatians: $P(\text{3 Dalmatians}|H_2) = (1/10)^3 = 10^{-3}$ (if there are 10 Dalmatians in the world) and $P(\text{3 Dalmatians}|H_3) = (1/100)^3 = 10^{-6}$ (if there are 100 dogs). The Dalmatian hypothesis is 10^3 times more likely than the dog hypothesis: **size principle**.

Prior. $P(H_k) =$ uniform over the hierarchy (or prior favouring middle levels).

0.34.2 3k.2 Posterior and Generalisation

The posterior probability of each hypothesis:

$$P(H_k|\mathbf{x}_{1:n}) = \frac{P(\mathbf{x}_{1:n}|H_k)P(H_k)}{\sum_j P(\mathbf{x}_{1:n}|H_j)P(H_j)}.$$

For $n = 1$ (one Dalmatian shown): all hypotheses H_1 through H_5 that include Dalmatians are equally supported (weak evidence). For $n = 3$ (three Dalmatians): the size principle strongly favours H_2 (Dalmatians) because the probability of sampling 3 Dalmatians by chance from the dog or animal category is tiny.

The generalisation probability for a new object y:

$$P(y \in \text{extension("dax")}|\mathbf{x}_{1:n}) = \sum_k P(H_k|\mathbf{x}_{1:n}) \cdot \mathbf{1}[y \in H_k].$$

Predictions. For $n = 1$ (one Dalmatian): uniform generalisation to all levels (weak Bayesian averaging). For $n = 3$ (three Dalmatians): strong generalisation to Dalmatians, weak to dogs, very weak to mammals. For $n = 1$ (one dog, one fish, one bird): strong generalisation to animals (the consistent hypothesis at a high level). These predictions match human data (Xu and Tenenbaum 2007) and are not predicted by prototype or exemplar models.

0.34.3 3k.3 Extensions: The Size Principle and PAC Learning

The size principle (larger hypotheses are less likely per example) is mathematically equivalent to **Occam's razor in hypothesis space**: for a fixed number of examples, smaller hypotheses (more restrictive, fewer consistent patterns) receive higher posterior probability than larger ones.

This connects to **PAC (Probably Approximately Correct) learning** (Valiant 1984): a concept class $\mathcal{H}$ is PAC-learnable if, for any $\epsilon, \delta > 0$, after $n \geq \frac{1}{\epsilon}(\log|\mathcal{H}| + \log(1/\delta))$ examples, the learner outputs a hypothesis with error $< \epsilon$ with probability $> 1-\delta$. The Bayesian learner with the size principle converges to the correct hypothesis at a rate governed by the concept's **VC dimension** — the maximum number of points that can be shattered (assigned arbitrary binary labels) by hypotheses in $\mathcal{H}$.

For the word learning hierarchy, the VC dimension is 1 (concepts are nested intervals in the taxonomy tree, and no two points can be shattered). The PAC bound gives $n = O(\log(1/\delta))$ examples for ϵ-accurate learning. This matches the empirical observation that children often correctly extend a new word after seeing 3–5 examples (**fast mapping**).

0.35 3j. Bayesian Hypothesis Testing vs Frequentist Testing

0.35.1 3j.1 The p-value and Its Limitations

The **p-value** $p = P(T(D) \geq T(D_{\text{obs}})|H_0)$ is the probability of observing a test statistic as extreme or more extreme than the observed value, given that the null hypothesis is true. Despite its ubiquity, the p-value has fundamental limitations as a measure of evidence:

1. **The p-value is not the probability of the null hypothesis.** $p = P(\text{data}|H_0)$, not $P(H_0|\text{data})$. The confusion of these two quantities is the **prosecutor's fallacy**.

2. **p-values depend on intentions.** If an experimenter plans to collect until $p < 0.05$, the type-I error rate is inflated; if they stop for other reasons, it is deflated. The same data give different p-values under different stopping rules.

3. **Significance threshold is arbitrary.** The threshold $\alpha = 0.05$ has no statistical justification. Lindley's paradox shows that for large samples, a test can be significant at $p < 0.05$ while the Bayes factor strongly favours the null.

Lindley's Paradox. For testing $H_0 : \mu = 0$ vs $H_1 : \mu \neq 0$ with n observations $\bar{x}$, the p-value for the z-test is $p = 2\Phi(-|\bar{x}|/\sigma_n)$ where $\sigma_n = \sigma/\sqrt{n}$. The Bayes factor for a normal prior $\mu \sim \mathcal{N}(0, \tau^2)$:

$$K_{01} = \frac{p(D|H_0)}{p(D|H_1)} = \sqrt{1 + n\tau^2/\sigma^2}\exp\left(-\frac{n\bar{x}^2}{2\sigma^2} \cdot \frac{n\tau^2/\sigma^2}{1 + n\tau^2/\sigma^2}\right).$$

For fixed p (say $p = 0.05$, meaning $|\bar{x}| = 1.96\sigma_n$) and large n:

$$K_{01} \approx \sqrt{n\tau^2/\sigma^2} \cdot \exp\left(-\frac{(1.96)^2}{2}\right) \to \infty \text{ as } n \to \infty.$$

The Bayes factor converges to infinity in favour of H_0 while the p-value stays fixed at 0.05! For large samples, the same marginally significant result is strong evidence for the null.

0.35.2 3j.2 The Bayes Factor: Properties and Computation

The Bayes factor $K = p(D|H_1)/p(D|H_0)$ is the natural Bayesian measure of evidence for one hypothesis over another. Its properties:

- **Transitivity**: $K_{12} = K_{13}/K_{23}$ (Bayes factors multiply transitively).
- **Symmetry**: $K_{10} = 1/K_{01}$.
- **Stopping rule invariance**: unlike p-values, Bayes factors do not depend on the experimenter's intentions. The same data yield the same Bayes factor regardless of the stopping rule.
- **Coherence**: combining two independent experiments gives a joint Bayes factor equal to the product of the individual Bayes factors: $K_{\text{joint}} = K_1 \cdot K_2$.

Computation for simple models. For the t-test comparison ($H_0 : \delta = 0$ vs $H_1 : \delta \sim \text{Cauchy}(0, r)$ where r is the scale parameter):

$$K_{10} = \frac{\int p(\bar{x}, s|\delta)\text{Cauchy}(\delta; 0, r)\, d\delta}{p(\bar{x}, s|\delta = 0)} = \int_0^\infty \left(1 + \frac{t^2/\nu}{(1+g/n)}\right)^{-(\nu+1)/2} \frac{(1+g)^{-1/2}}{(1+g/n)^{(\nu+1)/2}} \cdot \frac{1}{B(1/2, \nu/2)\sqrt{\nu}} dg,$$

where $t = \bar{x}\sqrt{n}/s$ is the t-statistic, $\nu = n - 1$ degrees of freedom, and $g = nr^2/s^2$ is the standardised prior width. This integral is numerically tractable and is implemented in `BayesFactor` (R), `pingouin` (Python), and JASP.

0.35.3 3j.3 Sequential Bayesian Testing: The Sequential Bayes Factor

The **Sequential Bayes Factor** (SBF; Schönbrodt et al., 2017) is a Bayesian alternative to sequential hypothesis testing. Collect data until the Bayes factor exceeds a threshold $K^+ > 1$ (evidence for H_1) or falls below $K^- < 1$ (evidence for H_0), or until a maximum sample size is reached:

$$\text{Stop and decide } H_1 \text{ if } K_n \geq K^+; \quad \text{Stop and decide } H_0 \text{ if } K_n \leq K^-;$$

$$\text{Continue collecting if } K^- < K_n < K^+.$$

The thresholds $K^+ = 10$ and $K^- = 1/10$ correspond to the Jeffreys scale's "strong evidence" criterion. The SBF is valid regardless of when you stop: unlike p-values, updating the Bayes factor after each observation does not inflate error rates. This property (**optional stopping is fine**) is a major practical advantage for cognitive science experiments, where early stopping can save resources if evidence accumulates quickly.

0.36 3i. Information Geometry of Learning and Memory

0.36.1 3i.1 Natural Gradient Descent

Standard gradient descent updates parameters in Euclidean space: $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}$. However, the parameter space of a probabilistic model is not Euclidean: it has the Fisher-Rao Riemannian metric (Section 3c). The **natural gradient** (Amari 1998) corrects for this curvature:

$$\theta \leftarrow \theta - \eta \mathcal{I}(\theta)^{-1} \nabla_\theta \mathcal{L},$$

where $\mathcal{I}(\theta)$ is the Fisher information matrix. The natural gradient is the steepest ascent direction in the Fisher-Rao metric, not Euclidean space. It is invariant under reparametrisation: if $\phi = g(\theta)$, the natural gradient for ϕ gives the same update trajectory as the natural gradient for θ.

Practical advantage. For exponential families, the natural gradient with respect to the natural parameters η coincides with the ordinary gradient with respect to the moment parameters $\mu = \mathbb{E}[T(X)]$:

$$\tilde{\nabla}_\eta \mathcal{L} = \mathcal{I}(\eta)^{-1} \nabla_\eta \mathcal{L} = \nabla_\mu \mathcal{L},$$

since $\mathcal{I}(\eta) = \nabla^2 A(\eta) = \partial\mu/\partial\eta$. In practice, natural gradient stochastic gradient descent (NGSGD) converges faster than standard SGD on models with highly non-Euclidean parameter spaces (RBMs, neural networks with Gaussian activations).

For Bayesian neural networks, the natural gradient in weight space corresponds to the **KFAC** (Kronecker-factored approximate curvature) approximation to the Fisher: $\mathcal{I}(W_l) \approx A_l \otimes S_l$, where $A_l = \mathbb{E}[\mathbf{a}_{l-1}\mathbf{a}_{l-1}^\top]$ and $S_l = \mathbb{E}[\delta_l \delta_l^\top]$ are the activation and gradient covariances at layer l. The Kronecker factorisation makes the $O(d^3)$ inversion of $\mathcal{I}$ tractable.

0.36.2 3i.2 Geodesics and Optimal Transport in Cognitive Change

Cognitive change (learning, forgetting, conceptual revision) can be modelled as a path through the statistical manifold $\mathcal{M}$ of probability distributions. Different types of change follow different geodesics:

Bayesian updating (conditioning on evidence D) follows the m**-geodesic** from prior p_0 to posterior $p_1 = p(\cdot|D)$. The m-geodesic in moment coordinates is a straight line: $\mu(t) = (1-t)\mu_0 + t\mu_1$, i.e., the moments of the distribution change linearly. This corresponds to a **mixture** of the prior and posterior: $p_t = (1-t)p_0 + tp_1$, which preserves the moment parameters.

Annealing (gradually cooling a temperature parameter) follows the e**-geodesic**: $p_t \propto p_0 \exp(t \log(p_1/p_0))$. This is an exponential family interpolation in natural parameter coordinates. Simulated annealing follows e-geodesics on the Boltzmann distribution, making the connection between algorithmic annealing and information geometry explicit.

Wasserstein distance and cognitive distance. The **Wasserstein-2 distance** (Earth Mover's Distance with quadratic cost) between distributions p and q on $\mathbb{R}^d$ is:

$$W_2(p,q)^2 = \inf_{\gamma \in \Gamma(p,q)} \int \|x - y\|^2 \, d\gamma(x,y),$$

where $\Gamma(p,q)$ is the set of couplings with marginals p and q. For Gaussian distributions $p = \mathcal{N}(\mu_1, \Sigma_1)$ and $q = \mathcal{N}(\mu_2, \Sigma_2)$:

$$W_2(p,q)^2 = \|\mu_1 - \mu_2\|^2 + \text{tr}(\Sigma_1 + \Sigma_2 - 2(\Sigma_1^{1/2}\Sigma_2\Sigma_1^{1/2})^{1/2}).$$

The Wasserstein distance metrises the space of distributions in a way that respects the geometry of the sample space (unlike KL divergence, which has no metric properties). For cognitive science, Wasserstein distance is a natural measure of "conceptual distance": two belief states are close if the beliefs differ mainly in location or spread, not in qualitative structure.

0.37 3h. Sequential Bayesian Updating and the Kalman Filter

0.37.1 3h.1 Online Bayesian Updating

In many cognitive and engineering applications, data arrive sequentially, and beliefs must be updated one observation at a time without storing all past data. Sequential Bayesian updating exploits the Markov property of the posterior: after observing $x_1, \ldots, x_n$, the posterior $p(\theta|x_{1:n})$ can be updated by x_{n+1} without re-examining $x_{1:n}$:

$$p(\theta|x_{1:n+1}) \propto p(x_{n+1}|\theta) \cdot p(\theta|x_{1:n}).$$

The posterior at step n acts as the prior for step $n+1$. This **recursive Bayes update** is the foundation of all online learning systems.

For exponential family likelihoods with conjugate priors, the update is simply a hyperparameter increment: $(\alpha_n, \beta_n) \mapsto (\alpha_n + T(x_{n+1}), \beta_n + 1)$ (for scalar hyperparameters). For a Beta-Bernoulli model, observing $x_{n+1} \in \{0, 1\}$:

$$(\alpha_n, \beta_n) \mapsto (\alpha_n + x_{n+1}, \beta_n + 1 - x_{n+1}).$$

After n observations with k successes: $p(\theta|x_{1:n}) = \text{Beta}(\alpha_0 + k, \beta_0 + n - k)$. The posterior mean is $(\alpha_0 + k)/(\alpha_0 + \beta_0 + n)$, interpolating between the prior mean $\alpha_0/(\alpha_0 + \beta_0)$ and the MLE k/n.

The virtue of online updating for cognitive modelling: it implements a **minimal memory** model of learning. The agent need not store past observations — only the current hyperparameter state (α_n, β_n) is needed for future inference. This is consistent with the empirical observation that humans integrate information across trials in a way that is well-approximated by an exponential moving average (a weighted sum with more weight on recent trials), which corresponds to online updating with a slight decay of the hyperparameters at each step.

0.37.2 3h.2 State-Space Models and the Kalman Filter

The **Kalman filter** (Kalman 1960) is the optimal Bayesian filter for linear-Gaussian state-space models:

$$\begin{aligned} \mathbf{x}_{t+1} &= A\mathbf{x}_t + \mathbf{w}_t, \quad \mathbf{w}_t \sim \mathcal{N}(0, Q), \\ \mathbf{y}_t &= H\mathbf{x}_t + \mathbf{v}_t, \quad \mathbf{v}_t \sim \mathcal{N}(0, R). \end{aligned}$$

The latent state $\mathbf{x}_t \in \mathbb{R}^d$ evolves according to linear dynamics with Gaussian process noise Q; the observation $\mathbf{y}_t \in \mathbb{R}^m$ is a linear function of the state corrupted by Gaussian measurement noise R.

The Kalman filter maintains the Gaussian posterior $p(\mathbf{x}_t|\mathbf{y}_{1:t}) = \mathcal{N}(\hat{\mathbf{x}}_{t|t}, P_{t|t})$ and updates it in two steps:

Predict step: Propagate the posterior forward through the dynamics:

$$\hat{\mathbf{x}}_{t+1|t} = A\hat{\mathbf{x}}_{t|t}, \qquad P_{t+1|t} = AP_{t|t}A^\top + Q.$$

Update step: Incorporate the new observation $\mathbf{y}_{t+1}$:

$$S_{t+1} = HP_{t+1|t}H^\top + R \quad \text{(innovation covariance)},$$

$$K_{t+1} = P_{t+1|t}H^\top S_{t+1}^{-1} \quad \text{(Kalman gain)},$$

$$\hat{\mathbf{x}}_{t+1|t+1} = \hat{\mathbf{x}}_{t+1|t} + K_{t+1}(\mathbf{y}_{t+1} - H\hat{\mathbf{x}}_{t+1|t}),$$
$$P_{t+1|t+1} = (I - K_{t+1}H)P_{t+1|t}.$$

The **innovation** $\mathbf{y}_{t+1} - H\hat{\mathbf{x}}_{t+1|t}$ is the prediction error: the difference between the observed $\mathbf{y}_{t+1}$ and the predicted observation $H\hat{\mathbf{x}}_{t+1|t}$. The Kalman gain K weights the innovation proportionally to the ratio of the state uncertainty $P_{t+1|t}$ to the total uncertainty S_{t+1}: large state uncertainty relative to measurement noise gives high Kalman gain (trust the observation more); small state uncertainty gives low Kalman gain (trust the prediction more).

Theorem 0.37.1 (Optimality of the Kalman filter)**.** The Kalman filter computes the exact Bayesian posterior $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ for the linear-Gaussian model. It minimises the mean squared estimation error $\mathbb{E}[\|\mathbf{x}_t - \hat{\mathbf{x}}_t\|^2]$ among all estimators, linear or nonlinear.

The Kalman filter is the continuous-time analogue of the conjugate Gaussian update: in both cases, the posterior is Gaussian with mean that is a precision-weighted average of the prior prediction and the observation, and variance that is reduced proportionally to the precision of the observation.

0.37.3 3h.3 Extensions: Extended and Unscented Kalman Filters

For nonlinear state-space models $\mathbf{x}_{t+1} = f(\mathbf{x}_t) + \mathbf{w}_t$, $\mathbf{y}_t = h(\mathbf{x}_t) + \mathbf{v}_t$, the posterior is no longer Gaussian. Two approximations are standard:

Extended Kalman Filter (EKF). Linearise f and h around the current estimate using first-order Taylor expansions:

$$f(\mathbf{x}) \approx f(\hat{\mathbf{x}}) + F(\mathbf{x} - \hat{\mathbf{x}}), \quad F = \nabla_{\mathbf{x}} f(\hat{\mathbf{x}}),$$

and proceed with the linear Kalman filter using the Jacobians F and $H = \nabla_{\mathbf{x}} h(\hat{\mathbf{x}})$. EKF is $O(d^2 m + d^3)$ per step (matrix-vector products plus inversion of $m \times m$ matrices). It fails when f or h are highly nonlinear and the estimation uncertainty is large.

Unscented Kalman Filter (UKF). Instead of linearising, propagate a set of **sigma points** $\mathcal{X} = \{\hat{\mathbf{x}}\} \cup \{\hat{\mathbf{x}} \pm \sqrt{(d+\lambda)P}\mathbf{e}_i\}_{i=1}^{d}$ (the columns of the Cholesky factor of the scaled covariance) through the nonlinear function:

$$\mathcal{Y}_i = f(\mathcal{X}_i), \qquad \hat{\mathbf{x}}^- = \sum_i W_i^m \mathcal{Y}_i, \qquad P^- = \sum_i W_i^c (\mathcal{Y}_i - \hat{\mathbf{x}}^-)(\mathcal{Y}_i - \hat{\mathbf{x}}^-)^\top + Q.$$

The UKF is exact for polynomial nonlinearities up to third order and generally more accurate than EKF at similar computational cost. For cognitive neuroscience, UKF is used to infer dynamical brain states from EEG/fMRI time series with nonlinear generative models.

0.38 3b. Exponential Families and Sufficient Statistics

0.38.1 3b.1 Sufficient Statistics and Fisher's Factorisation Theorem

In Bayesian inference, the prior and the posterior carry all probabilistic information about the parameter θ given data D. But not all features of the data are equally informative: many aspects of the raw data are irrelevant to the inference of θ, and only a lower-dimensional **summary** of the data — its **sufficient statistic** — matters.

Definition 0.38.1 (Sufficient statistic)**.** A statistic $T(D)$ is **sufficient** for parameter θ if the conditional distribution of the data given $T(D)$ does not depend on θ:

$$p(D \mid T(D), \theta) = p(D \mid T(D)).$$

Sufficiency means that $T(D)$ captures all the information in D relevant to θ: conditioning on T destroys no information about θ. Equivalently, θ and D are conditionally independent given $T(D)$: $D \perp\!\!\!\perp \theta \mid T(D)$.

Theorem 0.38.2 (Fisher-Neyman Factorisation)**.** $T(D)$ is sufficient for θ iff the likelihood can be written

$$p(D \mid \theta) = g(T(D), \theta) \cdot h(D)$$

for some functions g (depending on D only through T) and h (independent of θ).

Proof. ($\Rightarrow$) If T is sufficient, write $p(D|\theta) = p(D|T) \cdot p(T|\theta)$; since $p(D|T)$ does not depend on θ, set $h(D) = p(D|T(D))$ and $g(T(D), \theta) = p(T(D)|\theta)$. ($\Leftarrow$) If $p(D|\theta) = g(T, \theta)h(D)$, then $p(\theta|D) \propto g(T(D), \theta)p(\theta)$, so the posterior depends on D only through T. □

Example: Gaussian sufficient statistics. For i.i.d. $X_i \sim \mathcal{N}(\mu, \sigma^2)$ with known σ^2, the likelihood factors as:

$$p(\mathbf{x}|\mu) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\!\left(-\frac{(x_i-\mu)^2}{2\sigma^2}\right) = \underbrace{\exp\!\left(-\frac{n(\bar{x}-\mu)^2}{2\sigma^2}\right)}_{g(\bar{x},\mu)} \cdot \underbrace{\exp\!\left(-\frac{\sum(x_i-\bar{x})^2}{2\sigma^2}\right) \cdot (2\pi\sigma^2)^{-n/2}}_{h(\mathbf{x})}.$$

The sample mean $T(\mathbf{x}) = \bar{x} = \frac{1}{n}\sum x_i$ is sufficient for μ: the raw data $(x_1, \ldots, x_n)$ can be discarded and replaced by $\bar{x}$ for all inference about μ.

0.38.2 3b.2 The Exponential Family

Definition 0.38.3 (Exponential family)**.** A family of distributions is in **exponential form** if its density (w.r.t. base measure ν) can be written:

$$p(x \mid \eta) = h(x) \exp\!\Big(\eta^\top T(x) - A(\eta)\Big),$$

where $\eta \in \mathcal{H} \subseteq \mathbb{R}^k$ are **natural parameters**, $T : \mathcal{X} \to \mathbb{R}^k$ is the vector of **sufficient statistics**, $h(x) \geq 0$ is the carrier density, and $A(\eta) = \log \int h(x) e^{\eta^\top T(x)}\, d\nu(x)$ is the **log-partition function**.

The log-partition function $A(\eta)$ is **convex** on its domain $\mathcal{H}$ and plays a central role:

Proposition 0.38.4 (Cumulant generating property of A)**.**

$$\nabla_\eta A(\eta) = \mathbb{E}_{p(\cdot|\eta)}[T(X)], \qquad \nabla^2_\eta A(\eta) = \mathrm{Cov}_{p(\cdot|\eta)}(T(X)).$$

Proof. Differentiate $1 = \int h(x) \exp(\eta^\top T(x) - A(\eta))\, d\nu(x)$ under the integral sign. The first derivative gives $0 = \mathbb{E}[T(X)] - \nabla A(\eta)$, yielding $\nabla A = \mathbb{E}[T]$. The second derivative yields $\nabla^2 A = \mathbb{E}[TT^\top] - \mathbb{E}[T]\mathbb{E}[T]^\top = \mathrm{Cov}(T)$. □

Since $\mathrm{Cov}(T) \succeq 0$, A is convex. If A is strictly convex, the mapping $\eta \mapsto \mathbb{E}[T]$ is injective, establishing a

one-to-one correspondence between natural parameters and moment parameters $\mu = \mathbb{E}[T]$. This duality between η-parametrisation and μ-parametrisation (natural vs. moment) is the foundation of **information geometry**.

The Fisher information matrix for an exponential family is:

$$\mathcal{I}(\eta) = \nabla^2 A(\eta) = \mathrm{Cov}_\eta(T(X)).$$

The Fisher information matrix is the **Riemannian metric** on the statistical manifold: small perturbations $d\eta$ change the distribution by an amount (measured in KL divergence) of $\frac{1}{2} d\eta^\top \mathcal{I}(\eta) d\eta$. The geodesics of this metric are the **statistical geodesics**, and the exponential family has the special property that its η-coordinates are **affinely flat**: the curvature of the e-connection (exponential connection) is zero.

Examples of exponential families:

The **Bernoulli**: $p(x|\theta) = \theta^x (1-\theta)^{1-x}$ with $\eta = \log(\theta/(1-\theta))$ (log-odds), $T(x) = x$, $A(\eta) = \log(1 + e^\eta)$.

The **Gaussian** $\mathcal{N}(\mu, \sigma^2)$ with both parameters unknown: $\eta = (\mu/\sigma^2, -1/(2\sigma^2))$, $T(x) = (x, x^2)$, $A(\eta) = -\eta_1^2/(4\eta_2) - \frac{1}{2}\log(-2\eta_2) + \frac{1}{2}\log \pi$.

The **Dirichlet** $\mathrm{Dir}(\alpha_1, \ldots, \alpha_k)$: $\eta = \alpha - \mathbf{1}$, $T(x) = (\log x_1, \ldots, \log x_k)$, $A(\eta) = \sum_i \log \Gamma(\eta_i + 1) - \log \Gamma(\sum_i (\eta_i + 1))$.

0.38.3 3b.3 De Finetti's Representation Theorem

Theorem 0.38.5 (de Finetti 1937, extended)**.** A sequence of random variables $(X_1, X_2, \ldots)$ taking values in a Polish space $\mathcal{X}$ is **exchangeable** (the joint distribution of any finite sub-sequence is invariant under permutation of indices) if and only if there exists a unique probability measure μ on the space of probability measures $\mathcal{P}(\mathcal{X})$ such that:

$$p(x_1, \ldots, x_n) = \int_{\mathcal{P}(\mathcal{X})} \prod_{i=1}^{n} P(x_i)\, d\mu(P)$$

for all n.

The theorem says: every exchangeable sequence is a **mixture of i.i.d. sequences**. The mixing measure μ is uniquely determined by the sequence. The P that μ is supported on is the "type" of the sequence: the distribution that governs each X_i were the sequence i.i.d.

Proof sketch (binary case). For binary sequences, the key identity is that any symmetric function of $(X_1, \ldots, X_n)$ depends only on the empirical frequency $S_n = \sum_i X_i$. The measure μ is recovered as the limiting distribution of S_n/n: since (X_i) is exchangeable, $S_n/n \xrightarrow{a.s.} \theta$ for some $[0,1]$-valued random variable θ (by the backward martingale convergence theorem applied to S_n/n). The distribution of θ is μ. □

For cognitive science, de Finetti's theorem is foundational in two ways. First, it shows that Bayesian inference is not an arbitrary choice of prior but the *consequence* of exchangeability: an agent who treats observations symmetrically (not privileging any particular order) must represent its beliefs as a prior over parameters. The prior is forced by the symmetry. Second, the theorem provides a non-parametric model

for categorical data: the Dirichlet process is the conjugate prior in the de Finetti representation when (X_i) is exchangeable with $\mathcal{X} = \{1, \ldots, K\}$.

0.39 3c. Information Geometry: The Riemannian Geometry of Statistics

0.39.1 3c.1 The Fisher-Rao Metric

The space of probability distributions over a fixed sample space $\mathcal{X}$, parametrised by $\theta \in \Theta$, carries a natural Riemannian metric: the **Fisher information metric**. For a parametric family $\{p(x|\theta)\}_\theta$, the Fisher information matrix is:

$$g_{ij}(\theta) = \mathcal{I}(\theta)_{ij} = \mathbb{E}_{p(\cdot|\theta)}\left[\frac{\partial \log p(X|\theta)}{\partial \theta_i}\frac{\partial \log p(X|\theta)}{\partial \theta_j}\right] = -\mathbb{E}\left[\frac{\partial^2 \log p(X|\theta)}{\partial \theta_i \partial \theta_j}\right].$$

This is a positive semidefinite matrix (generically positive definite) that defines a Riemannian metric on the parameter space Θ.

The metric g_{ij} has a privileged status among all Riemannian metrics on Θ: by **Chentsov's theorem** (1982), the Fisher metric is the unique (up to scalar multiples) Riemannian metric on the simplex of discrete distributions that is invariant under sufficient statistics (i.e., unchanged by any sufficient transformation of the data space). Sufficiency-invariant metrics measure the intrinsic information geometry, independent of how data are represented.

The **Cramér-Rao bound** is an immediate consequence: for any unbiased estimator $\hat{\theta}(X)$ of θ,

$$\mathrm{Cov}(\hat{\theta}) \succeq \mathcal{I}(\theta)^{-1}.$$

The inverse Fisher information matrix is the minimum achievable covariance. Estimators achieving this bound — **efficient estimators** — lie at the heart of optimal statistical inference.

0.39.2 3c.2 Dual Connections and the α-Geometry

Information geometry (Amari 1985, 2016) studies the statistical manifold $\mathcal{M} = \{p(\cdot|\theta)\}_\theta$ with the Fisher metric and a family of α**-connections** $\nabla^{(\alpha)}$ parametrised by $\alpha \in \mathbb{R}$.

The α-connection has Christoffel symbols:

$$\Gamma^{(\alpha)}_{ij,k} = \mathbb{E}\left[\left(\partial_i\partial_j\ell + \frac{1-\alpha}{2}\partial_i\ell\partial_j\ell\right)\partial_k\ell\right],$$

where $\ell = \log p(x|\theta)$ is the log-likelihood and $\partial_i = \partial/\partial\theta_i$. The special cases are:

- $\alpha = 1$: the **exponential connection** $\nabla^{(e)}$, under which exponential families are flat (zero curvature).
- $\alpha = -1$: the **mixture connection** $\nabla^{(m)}$, under which mixture families are flat.
- $\alpha = 0$: the Levi-Civita connection of the Fisher metric.

The e-connection and m-connection are **dual**: $\nabla^{(-1)} = (\nabla^{(1)})^*$ with respect to the Fisher metric. This duality is the geometric expression of the fact that the KL divergence $\mathrm{KL}(p\|q)$ is not symmetric: it is the

e-divergence in one direction and the m-divergence in the other. The information-geometric projection theorem states: the m-projection of a distribution onto an exponential family equals the distribution in the family with the same sufficient statistics (the moment-matching estimate); the e-projection minimises KL divergence.

For variational inference, the variational family $\mathcal{Q}$ is typically an exponential family (e.g., a Gaussian or mean-field factorisation), and the ELBO maximisation is the m-projection of the posterior $p(\theta|D)$ onto $\mathcal{Q}$ under the reverse KL: $q^* = \arg\min_{q \in \mathcal{Q}} \mathrm{KL}(q\|p) =$ the m-projection. The forward KL minimisation $\arg\min_{q \in \mathcal{Q}} \mathrm{KL}(p\|q)$ is the e-projection, corresponding to expectation propagation.

Section *II*

Section II — Deep Bayesian Inference

0.40 4. The Intractability of Exact Inference

The Bayesian framework is conceptually elegant and logically compelling. It is also, in general, computationally intractable. The posterior distribution $p(\theta \mid D) = p(D \mid \theta)p(\theta)/p(D)$ requires evaluating the evidence $p(D) = \int p(D \mid \theta)p(\theta)\, d\theta$. For high-dimensional θ, this integral is over a space of millions or billions of dimensions, and no analytical formula is available.

This is not a minor technical obstacle. It is a fundamental barrier. A neural network with 10^6 parameters defines a 10^6-dimensional posterior whose exact computation is, in principle, as hard as enumerating all possible states of a system with 10^6 continuous degrees of freedom. The Bayesian Brain hypothesis requires not just that the brain performs Bayesian inference in principle, but that it does so approximately and efficiently in real time.

This Section develops the two primary families of approximate inference algorithms: **Variational Bayes** (optimisation-based approximation) and **Markov Chain Monte Carlo** (sampling-based approximation). Together, they make deep Bayesian inference computationally feasible, and they form the algorithmic backbone of probabilistic machine learning.

0.41 5. Variational Inference

0.41.1 5.1 The Variational Problem

Variational inference converts the problem of computing a posterior into an optimisation problem. The idea is to choose a **variational family** $\mathcal{Q} = \{q_\lambda(\theta)\}_\lambda$ of tractable distributions, parametrised by variational parameters λ, and find the member of $\mathcal{Q}$ closest to the true posterior $p(\theta \mid D)$.

Closeness is measured by the **Kullback-Leibler divergence**. For distributions q and p on the same space:

$$\mathrm{KL}(q\|p) = \int q(\theta) \log \frac{q(\theta)}{p(\theta)}\, d\theta = \mathbb{E}_{q(\theta)}\left[\log \frac{q(\theta)}{p(\theta)}\right] \geq 0.$$

The non-negativity, known as Gibbs' inequality, follows from Jensen's inequality applied to the convex function $-\log$: $-\mathbb{E}_q[\log(p/q)] \geq -\log \mathbb{E}_q[p/q] = -\log 1 = 0$, with equality iff $q = p$ almost everywhere. The KL divergence is not symmetric: $\mathrm{KL}(q\|p) \neq \mathrm{KL}(p\|q)$ in general.

The variational objective is $q^*_\lambda = \arg\min_\lambda \mathrm{KL}(q_\lambda \| p(\cdot \mid D))$. But this requires knowing $p(\theta \mid D)$ to compute the KL, which is what we want to find. The key identity resolves this circularity. Writing $p(\theta \mid D) = p(D \mid \theta)p(\theta)/p(D)$:

$$\mathrm{KL}(q_\lambda \| p(\cdot \mid D)) = \mathbb{E}_{q_\lambda}[\log q_\lambda(\theta)] - \mathbb{E}_{q_\lambda}[\log p(D \mid \theta)p(\theta)] + \log p(D).$$

Rearranging:

$$\log p(D) = \underbrace{\mathbb{E}_{q_\lambda}[\log p(D \mid \theta)p(\theta)] - \mathbb{E}_{q_\lambda}[\log q_\lambda(\theta)]}_{\mathcal{L}(\lambda)} + \mathrm{KL}(q_\lambda \| p(\cdot \mid D)).$$

Since $\mathrm{KL} \geq 0$, the term $\mathcal{L}(\lambda)$ is a lower bound on $\log p(D)$. This is the **Evidence Lower BOund** (ELBO). Because $\log p(D)$ does not depend on λ, minimising $\mathrm{KL}(q_\lambda \| p(\cdot \mid D))$ is equivalent to maximising $\mathcal{L}(\lambda)$.

The ELBO can be written in a more revealing form. Let $p(D, \theta) = p(D \mid \theta)p(\theta)$ be the joint distribution. Then:

$$\mathcal{L}(\lambda) = \mathbb{E}_{q_\lambda(\theta)}[\log p(D, \theta)] - \mathbb{E}_{q_\lambda(\theta)}[\log q_\lambda(\theta)] = \mathbb{E}_{q_\lambda}[\log p(D \mid \theta)] - \mathrm{KL}(q_\lambda \| p(\cdot)).$$

The first term rewards the variational distribution for placing mass on parameters that explain the data well. The second term penalises the variational distribution for deviating from the prior. This is the variational expression of Occam's razor: the optimal approximation balances data fit against prior conformity.

0.41.2 5.1b The ELBO: Complete Derivation

The ELBO identity is worth deriving in full. Starting from the log marginal likelihood:

$$\log p(D) = \log \int p(D, \theta)\, d\theta.$$

Multiplying and dividing by $q_\lambda(\theta)$ inside the integral:

$$\log p(D) = \log \int q_\lambda(\theta)\, \frac{p(D, \theta)}{q_\lambda(\theta)}\, d\theta \geq \int q_\lambda(\theta)\, \log \frac{p(D, \theta)}{q_\lambda(\theta)}\, d\theta = \mathcal{L}(\lambda),$$

where the inequality is Jensen's applied to the concave log function. Expanding $\mathcal{L}$:

$$\mathcal{L}(\lambda) = \mathbb{E}_{q_\lambda}[\log p(D, \theta)] - \mathbb{E}_{q_\lambda}[\log q_\lambda(\theta)] = \mathbb{E}_{q_\lambda}[\log p(D|\theta)] - \mathrm{KL}(q_\lambda(\theta) \| p(\theta)).$$

The first term rewards the variational distribution for assigning mass to parameters with high likelihood; the second term penalises deviation from the prior. The gap between $\log p(D)$ and $\mathcal{L}(\lambda)$ is exactly $\mathrm{KL}(q_\lambda \| p(\cdot|D)) \geq 0$, so maximising $\mathcal{L}$ simultaneously minimises this KL.

The ELBO is also interpretable as **expected log-likelihood minus regularisation**: maximising it trades off data fit (pushing q toward high-likelihood regions) against prior adherence (pulling q back toward $p(\theta)$). In the free-energy language of Section IV, $-\mathcal{L}(\lambda)$ is exactly the variational free energy $\mathcal{F}(q, D)$.

0.41.3 5.2 The Mean-Field Approximation

The most widely used variational family is the **mean-field** family, which assumes the variational distribution factorises across disjoint groups of parameters:

$$q_\lambda(\theta) = \prod_{i=1}^{m} q_{\lambda_i}(\theta_i).$$

The parameters $\theta_1, \ldots, \theta_m$ are treated as independent under the variational posterior, even if they are dependent under the true posterior. This independence assumption is the approximation: it converts a complex multivariate posterior into a product of marginals, which is tractable to optimise.

Under the mean-field factorisation, the ELBO-optimal form of each factor $q_{\lambda_i}(\theta_i)$ can be derived in closed form. Taking the functional derivative of $\mathcal{L}$ with respect to q_i (treating all other factors as fixed) and setting it to zero:

$$\log q_i^*(\theta_i) = \mathbb{E}_{q_{-i}}[\log p(D, \theta)] + \text{const},$$

where the expectation is over all variables except θ_i, and the constant ensures normalisation. This defines

each optimal factor as proportional to the exponentiated expected log joint, where the expectation is taken over the other factors.

The **coordinate ascent variational inference** (CAVI) algorithm iterates these updates cyclically: update q_1 with $q_2, \ldots, q_m$ held fixed; then update q_2 with $q_1, q_3, \ldots, q_m$ held fixed; and so on, until the ELBO converges. Convergence to a local maximum of the ELBO is guaranteed by the monotone improvement property of coordinate ascent.

0.41.4 5.3 Normalising Flows: Expressive Variational Families

The mean-field approximation sacrifices accuracy for tractability. More expressive variational families can be constructed using **normalising flows**: sequences of invertible, differentiable transformations that deform a simple base distribution into a complex one.

Let $z_0 \sim q_0(z_0)$ be a simple base distribution (e.g., a standard Gaussian). A normalising flow applies a sequence of K invertible transformations $f_1, f_2, \ldots, f_K$, each parametrised by learnable weights, to produce $z_K = f_K \circ \cdots \circ f_1(z_0)$. By the change-of-variables formula for densities:

$$\log q_K(z_K) = \log q_0(z_0) - \sum_{k=1}^{K} \log \left| \det J_{f_k}(z_{k-1}) \right|,$$

where J_{f_k} is the Jacobian matrix of f_k. The density of z_K is therefore computable as long as the determinant of each Jacobian is computable.

The design of normalising flows is an active area that trades off expressiveness (the complexity of distributions that can be represented) against tractability (the cost of computing Jacobian determinants). **Planar flows** use transformations of the form $f(z) = z + uh(w^\top z + b)$ with $u, w \in \mathbb{R}^d$ and scalar b; their Jacobians have determinants computable in $O(d)$ time by the matrix determinant lemma. **Autoregressive flows** use transformations where each coordinate depends only on previous coordinates, yielding triangular Jacobians with trivially computable determinants. **Continuous normalising flows**, based on neural ordinary differential equations, parametrise the transformation as the solution of a differential equation, enabling more flexible geometries at the cost of numerical integration.

Normalising flows connect to the mathematics of mind through the hypothesis that cortical representations are coordinates in a latent space related to sensory data by a learned invertible transformation — a flow that inverts the generative process of the world.

0.41.5 5.4 The Variational Autoencoder

The **variational autoencoder** (VAE) is the canonical deep Bayesian model: a generative model of observed data x through a latent variable z, with inference performed by a learned variational approximation.

The generative model specifies a prior $p(z)$ (typically $\mathcal{N}(0, I)$) and a likelihood $p_\theta(x \mid z)$ parametrised by a neural network (the **decoder**). The inference model is a variational distribution $q_\phi(z \mid x)$ also parametrised by a neural network (the **encoder**). The ELBO is:

$$\mathcal{L}(\theta, \phi; x) = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x \mid z)] - \mathrm{KL}(q_\phi(z \mid x) \| p(z)).$$

The first term is the expected reconstruction quality: it rewards the decoder for recovering x from samples

of the encoded latent. The second term is the KL from the variational posterior to the prior: it regularises the latent space, preventing posterior collapse and encouraging a smooth latent geometry.

Training by gradient descent requires differentiating through the sampling operation $z \sim q_\phi(z \mid x)$, which is not directly differentiable. The **reparametrisation trick** resolves this: write $z = \mu_\phi(x) + \sigma_\phi(x) \odot \epsilon$ where $\epsilon \sim \mathcal{N}(0, I)$ and μ_ϕ, σ_ϕ are the encoder's outputs. The randomness is now in ϵ, which does not depend on ϕ, so gradients can flow through μ_ϕ and σ_ϕ by standard backpropagation.

The VAE is a mathematical model of the generative hypothesis of perception: the brain maintains a generative model of the world, infers latent causes of sensory data, and generates predictions that are compared to actual input. This hypothesis, articulated in different vocabularies by Helmholtz, Hinton, and Friston, is the foundation of the next Section .

0.42 5c. The Expectation-Maximisation Algorithm

0.42.1 5c.1 Incomplete Data and Latent Variables

Many generative models involve **latent (hidden) variables** z that are not directly observed. The observed data x are modelled as having been generated by first sampling $z \sim p(z|\theta)$ and then sampling $x|z \sim p(x|z, \theta)$. The **complete-data log-likelihood** $\log p(x, z|\theta)$ would be easy to maximise over θ, but the **incomplete-data log-likelihood** $\log p(x|\theta) = \log \int p(x, z|\theta)\, dz$ involves an intractable integral.

The **Expectation-Maximisation (EM) algorithm** (Dempster, Laird, Rubin 1977) maximises $\log p(x|\theta)$ by iterating two steps:

- **E step:** Compute the **expected complete-data log-likelihood** under the current posterior over latent variables:
$$Q(\theta, \theta^{\text{old}}) = \mathbb{E}_{p(z|x,\theta^{\text{old}})}[\log p(x, z|\theta)]\,.$$
- **M step:** Maximise Q over θ:
$$\theta^{\text{new}} = \arg\max_\theta Q(\theta, \theta^{\text{old}}).$$

Theorem 0.42.1 (EM monotone improvement)**.** Each EM iteration does not decrease the incomplete-data log-likelihood:
$$\log p(x|\theta^{\text{new}}) \geq \log p(x|\theta^{\text{old}}).$$

Proof. The Jensen-based decomposition:
$$\log p(x|\theta) = Q(\theta, \theta^{\text{old}}) - \text{KL}(p(z|x, \theta^{\text{old}}) \| p(z|x, \theta)) + \text{const.}$$
In the M step, $Q(\theta^{\text{new}}, \theta^{\text{old}}) \geq Q(\theta^{\text{old}}, \theta^{\text{old}})$ (by maximisation). Since $\text{KL} \geq 0$ and $\text{KL} = 0$ at $\theta = \theta^{\text{old}}$, the decrease in KL when moving from θ^{old} to θ^{new} cannot outweigh the increase in Q. Formally:
$$\log p(x|\theta^{\text{new}}) \geq Q(\theta^{\text{new}}, \theta^{\text{old}}) - \text{KL}(\cdot\|\cdot)|_{\theta^{\text{new}}} \geq Q(\theta^{\text{old}}, \theta^{\text{old}}) - 0 = \log p(x|\theta^{\text{old}}). \quad \square$$

EM converges to a local maximum (or saddle point) of $\log p(x|\theta)$; convergence to the global maximum is not guaranteed. The convergence rate is linear, with rate given by the **fraction of missing information**:

$r = 1 - \mathcal{I}_{\text{obs}}/\mathcal{I}_{\text{comp}}$, the ratio of observed Fisher information to complete Fisher information. Large missing information (high latent variable uncertainty) implies slow convergence.

0.42.2 5c.2 The Gaussian Mixture Model

The canonical application of EM is the **Gaussian Mixture Model** (GMM). The generative model: component k is chosen with probability π_k ($\sum_k \pi_k = 1$), then $x|k \sim \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$. The full parameter set is $\theta = (\boldsymbol{\pi}, \boldsymbol{\mu}_{1:K}, \boldsymbol{\Sigma}_{1:K})$.

E step. Compute the **responsibilities** $r_{nk} = p(z_n = k|x_n, \theta^{\text{old}})$:

$$r_{nk} = \frac{\pi_k^{\text{old}} \mathcal{N}(x_n; \boldsymbol{\mu}_k^{\text{old}}, \boldsymbol{\Sigma}_k^{\text{old}})}{\sum_{j=1}^{K} \pi_j^{\text{old}} \mathcal{N}(x_n; \boldsymbol{\mu}_j^{\text{old}}, \boldsymbol{\Sigma}_j^{\text{old}})}.$$

M step. Update parameters using the responsibilities:

$$N_k = \sum_{n=1}^{N} r_{nk}, \qquad \pi_k^{\text{new}} = \frac{N_k}{N},$$

$$\boldsymbol{\mu}_k^{\text{new}} = \frac{1}{N_k} \sum_n r_{nk} x_n, \qquad \boldsymbol{\Sigma}_k^{\text{new}} = \frac{1}{N_k} \sum_n r_{nk} (x_n - \boldsymbol{\mu}_k^{\text{new}})(x_n - \boldsymbol{\mu}_k^{\text{new}})^\top.$$

These update equations have an intuitive interpretation: the new mean of component k is the responsibility-weighted average of all data points, and the new covariance is the responsibility-weighted sample covariance. The EM algorithm for GMMs is a **soft k-means**: it assigns each point fractionally to each cluster (via r_{nk}) rather than hard-assigning to the nearest centroid. Hard k-means is the $T \to 0$ limit of EM with isotropic equal-variance Gaussians, where responsibilities become $\{0, 1\}$-valued.

0.42.3 5c.3 EM as Coordinate Ascent in the ELBO

EM has a variational interpretation: it is coordinate ascent in the ELBO $\mathcal{L}(q, \theta) = \mathbb{E}_q[\log p(x, z|\theta)] - \mathbb{E}_q[\log q(z)]$ alternating between:

- **E step:** Optimise over q (with θ fixed) $\Rightarrow q^* = p(z|x, \theta)$ (exact posterior).
- **M step:** Optimise over θ (with q fixed) $\Rightarrow \theta^* = \arg\max_\theta \mathbb{E}_{q^*}[\log p(x, z|\theta)]$.

The key difference from variational Bayes (Section 5.2): the E step uses the *exact* posterior $q^* = p(z|x, \theta)$ (not an approximation from a restricted family). EM is applicable when the exact E step is tractable; variational Bayes is used when it is not.

Generalised EM: when the exact E step is intractable (e.g., for non-conjugate models), one can use an approximate E step (variational EM) or a partial M step (gradient EM, where the M step does not go to the maximum but takes a gradient step). These generalisations preserve the monotone improvement property as long as the ELBO does not decrease in either step.

0.43 5h. Expectation Propagation: Theory and Cognitive Applications

0.43.1 5h.1 Expectation Propagation Algorithm

Expectation Propagation (EP; Minka 2001) is an approximate inference algorithm that approximates a distribution $p(\theta) \propto \prod_i f_i(\theta)$ by finding a distribution $q(\theta) = \prod_i \tilde{f}_i(\theta)$ in a tractable family (typically an exponential family) such that q best approximates p in a local, factor-by-factor sense.

The algorithm iterates:

1. **Cavity computation**: remove factor i from q: $q_{-i}(\theta) \propto q(\theta)/\tilde{f}_i(\theta)$.
2. **Tilted distribution**: form $\hat{p}_i(\theta) \propto f_i(\theta) q_{-i}(\theta)$.
3. **Moment matching**: find $\tilde{f}_i^{\text{new}}(\theta)$ such that $q_{-i}(\theta)\tilde{f}_i^{\text{new}}(\theta)$ matches the moments of $\hat{p}_i$:

$$\mathbb{E}_{q_{-i}\tilde{f}_i^{\text{new}}}[T(\theta)] = \mathbb{E}_{\hat{p}_i}[T(\theta)],$$

   where $T(\theta)$ is the vector of sufficient statistics.

For an exponential family with sufficient statistics T, the moment matching step amounts to finding the natural parameters $\tilde{\eta}_i$ of $\tilde{f}_i$ such that $\nabla A(\eta_q) = \mathbb{E}_{\hat{p}_i}[T]$ (where A is the log-partition function and η_q are the natural parameters of q). For Gaussian q, this means matching the mean and variance of $\hat{p}_i$.

EP fixed points. EP converges to a fixed point characterised by: $\mathbb{E}_{q_{-i}\tilde{f}_i}[T] = \mathbb{E}_{\hat{p}_i}[T]$ for all i. These fixed points are the stationary points of the **Bethe free energy**:

$$\mathcal{F}_B = -\sum_i \mathbb{E}_q[\log f_i(\theta)] + \text{KL}(q\|p_0) + \sum_i \text{KL}(q\|\hat{p}_i),$$

where p_0 is the base measure. EP minimises the Bethe free energy in a block coordinate-ascent fashion, alternating between the site approximations $\tilde{f}_i$.

0.43.2 5h.2 EP for Robust Regression

For a regression model with outlier-robust likelihood $p(y|f) = \text{Student-}t(y; f, \sigma^2, \nu)$ (with degrees of freedom ν controlling outlier robustness), the likelihood does not factorise into a Gaussian form, preventing conjugate updating. EP approximates each likelihood factor by a Gaussian:

$$\tilde{f}_i(f_i) = \mathcal{N}(f_i; \tilde{\mu}_i, \tilde{\sigma}_i^2).$$

The EP update for site i requires computing the moments of $\hat{p}_i \propto \text{Student-}t(y_i; f_i, \sigma^2, \nu) \cdot \mathcal{N}(f_i; \mu_{-i}, \sigma_{-i}^2)$. This product is a truncated or extended Gaussian depending on the Student-t parameters; its moments can be computed via one-dimensional numerical integration. The Gaussian $\tilde{f}_i$ is then matched to these moments.

Cognitive application. Students learning from teachers receive feedback that may be inaccurate or misleading (analogous to outliers). A Bayesian learner with a Student-t likelihood (or EP approximation to it) downweights surprising feedback (high prediction error, corresponding to heavy-tailed likelihood), maintaining robustness to noisy or adversarial teaching. EP provides a computationally efficient implementation of this robust Bayesian learning.

0.44 14i. Influence Diagrams and Decision-Theoretic Reasoning

0.44.1 14i.1 Influence Diagrams

An **influence diagram** (Howard and Matheson 1984) extends a Bayesian network with decision nodes (representing the agent's choices) and a utility node (representing the agent's objectives). The structure encodes the temporal and informational order of decisions and observations.

A simple influence diagram has:

- **Chance nodes** C_i: random variables with conditional probability tables.
- **Decision nodes** D_j: deterministic choices made by the agent.
- **Utility node** U: the agent's payoff as a function of parent nodes.
- **Information sets**: arrows into decision nodes D_j from chance nodes observed before D_j.

The **maximum expected utility** (MEU) policy is the decision rule π that maximises the expected utility:

$$\pi^* = \arg\max_{\pi} \mathbb{E}_{\pi}[U].$$

For a decision sequence $D_1, D_2, \ldots$ with observations between decisions, the MEU is computed by **backward induction**: starting from the last decision and working backward, computing the optimal decision at each step given the optimal decisions at future steps.

0.44.2 14i.2 POMDPs as Influence Diagrams

A POMDP (Section 9b) can be unrolled into an influence diagram: chance nodes are the hidden states S_t and observations O_t; decision nodes are the actions A_t; utility nodes are the rewards R_t. The agent's observation history up to time t is the information set of decision A_t.

The key quantity is the **belief state** $b_t = P(S_t | O_{1:t}, A_{1:t-1})$: a sufficient statistic for the agent's information at time t. The POMDP reduces to an MDP over belief states: $b_{t+1} = \tau(b_t, A_t, O_{t+1})$ (the belief update operator) and the value function $V^*(b)$ over beliefs satisfies the POMDP Bellman equation (Section 9b).

The influence diagram representation makes the **value of information** explicit: the value of obtaining additional observation O before making decision D is $\text{VOI}(O; D) = \mathbb{E}_O[\max_d \mathbb{E}[U|O, d]] - \max_d \mathbb{E}[U|d]$. This is the expected gain from making a better-informed decision, and it provides a decision-theoretic justification for Active Inference's epistemic value term (Section 9b).

0.45 16i. Modal Logic and Epistemic Accessibility

0.45.1 16i.1 Normal Modal Logics

Modal logic extends propositional logic with operators $\Box$ (necessity, "it is necessary that") and $\Diamond$ (possibility, "it is possible that"). The **possible worlds semantics** (Kripke 1959, 1963): a model $\mathcal{M} = (W, R, V)$ consists of a set of possible worlds W, an accessibility relation $R \subseteq W \times W$, and a valuation

$V : W \times P \to \{0, 1\}$ assigning truth values to propositions at worlds. Semantics:

$$\mathcal{M}, w \vDash \Box\varphi \iff \forall w' : wRw' \Rightarrow \mathcal{M}, w' \vDash \varphi,$$

$$\mathcal{M}, w \vDash \Diamond\varphi \iff \exists w' : wRw' \wedge \mathcal{M}, w' \vDash \varphi.$$

Different properties of R give different logics:

- R reflexive: $K4$ axiom T: $\Box\varphi \Rightarrow \varphi$ (what is necessary is true).
- R transitive: $K4$ axiom 4: $\Box\varphi \Rightarrow \Box\Box\varphi$ (positive introspection).
- R Euclidean ($(wRu \wedge wRv) \Rightarrow uRv$): axiom 5: $\Diamond\varphi \Rightarrow \Box\Diamond\varphi$ (negative introspection).
- R reflexive + transitive: logic $S4$ (the logic of provability).
- R reflexive + transitive + symmetric: logic $S5$ (the logic of knowledge).

Epistemic logic uses K_i ("agent i knows that") with $S5$ as the default: knowledge is veridical ($K_i\varphi \Rightarrow \varphi$), positively introspective ($K_i\varphi \Rightarrow K_iK_i\varphi$), and negatively introspective ($\neg K_i\varphi \Rightarrow K_i\neg K_i\varphi$). These postulates are the epistemic analogues of the Kolmogorov axioms for probability.

0.45.2 16i.2 Doxastic Logic and Belief

Doxastic logic (Hintikka 1962) uses B_i ("agent i believes that") with a weaker logic than $S5$: beliefs need not be veridical ($B_i\varphi \not\Rightarrow \varphi$). The standard doxastic logic KD45 includes:

- Axiom D: $B_i\varphi \Rightarrow \neg B_i\neg\varphi$ (beliefs are consistent).
- Axiom 4: $B_i\varphi \Rightarrow B_iB_i\varphi$ (positive introspection for beliefs).
- Axiom 5: $\neg B_i\varphi \Rightarrow B_i\neg B_i\varphi$ (negative introspection for beliefs).

Doxastic logic models rational agents with possibly false beliefs. The connection to probability: the probability of φ given the agent's beliefs can be represented as $P_i(\varphi) = P(\varphi | B_i\varphi_1, \ldots, B_i\varphi_n)$ where $\varphi_1, \ldots, \varphi_n$ are the propositions the agent believes. Probabilistic doxastic logic (Fagin and Halpern 1994) extends this to real-valued belief degrees, bridging the Boolean and probabilistic approaches to belief.

0.46 22h. Formal Theories of Discourse

0.46.1 22h.1 Discourse Representation Theory

Discourse Representation Theory (DRT; Kamp 1981; Kamp and Reyle 1993) is a formal semantic theory of discourse that represents the meaning of multi-sentence texts as **Discourse Representation Structures** (DRSs). A DRS $K = (U_K, C_K)$ consists of:

- A set of **discourse referents** U_K (variables for entities mentioned in the discourse).
- A set of **conditions** C_K (formulas relating the discourse referents).

For a text "John owns a dog. It is brown.":

- After "John owns a dog": $U_K = \{j, d\}$, $C_K = \{\text{John}(j), \text{dog}(d), \text{owns}(j, d)\}$.

- After "It is brown": $U_K = \{j, d\}$, $C_K = \{\text{John}(j), \text{dog}(d), \text{owns}(j, d), \text{brown}(d)\}$.

The pronoun "it" is resolved to the discourse referent d introduced in the first sentence.

DRS update. Processing each sentence updates the DRS: new discourse referents are added to U_K; new conditions are added to C_K; pronouns are resolved to existing discourse referents (accessibility conditions ensure that only referents in the right structural position can be antecedents). The DRS is the formal semantics's counterpart of the psycholinguist's **situation model**: the mental representation of the state of affairs described by the text.

0.47 5g. Variational EM and Bayesian Model Selection

0.47.1 5g.1 The Evidence Lower Bound as a Model Selection Criterion

In Bayesian model selection, we want to compare models $M_1, M_2, \ldots$ for the same data D. The optimal criterion is the marginal likelihood $p(D|M_k) = \int p(D|\theta, M_k)p(\theta|M_k)\, d\theta$. For complex models, this integral is intractable. The ELBO $\mathcal{L}(q, M_k)$ is a lower bound on $\log p(D|M_k)$, and can be used as a surrogate criterion.

Variational model selection: choose $M^* = \arg\max_{M_k} \max_{q \in \mathcal{Q}} \mathcal{L}(q, M_k)$. This is the Bayesian Information Criterion (BIC) approximation when $\mathcal{Q}$ is the point mass at the MAP: $\mathcal{L}(\delta_{\hat{\theta}}, M_k) = \log p(D|\hat{\theta}, M_k) - \frac{k}{2} \log n$ (where k is the number of parameters and n is the number of data points). The BIC penalises model complexity via the $-\frac{k}{2} \log n$ term.

For the full variational ELBO (not the MAP approximation), model selection is more nuanced: the ELBO automatically penalises model complexity through the KL term $\text{KL}(q(\theta) \| p(\theta|M_k))$. Models with more parameters that are poorly identified by the data (flat likelihood over a large parameter space) have high KL, reducing the ELBO. This is the variational analogue of Bayesian Occam's razor: the ELBO selects the model that best compresses the data, balancing fit and complexity.

0.47.2 5g.2 Variational EM for Mixture Models

The **Variational EM** (VEM) algorithm for Bayesian mixture models treats both the mixture assignments $\mathbf{z}$ and the parameters $\boldsymbol{\theta} = (\pi, \mu_{1:K}, \Lambda_{1:K})$ as latent variables, placing priors on $\boldsymbol{\theta}$ and performing variational inference over both.

For a Bayesian GMM with Dirichlet prior $\pi \sim \text{Dir}(\alpha_0 \mathbf{1})$, Gaussian-Wishart prior $(\mu_k, \Lambda_k) \sim \mathcal{NW}(\mathbf{m}_0, \beta_0, \nu_0, \mathbf{W}_0)$, and latent assignments $z_n \sim \text{Cat}(\pi)$:

E step (update variational assignments $q(\mathbf{z}) = \prod_n q(z_n)$):

$$\ln \rho_{nk} = \mathbb{E}[\ln \pi_k] + \frac{1}{2}\mathbb{E}[\ln |\Lambda_k|] - \frac{D}{2} \ln(2\pi) - \frac{1}{2}\mathbb{E}_{\mu_k, \Lambda_k}[(\mathbf{x}_n - \mu_k)^\top \Lambda_k (\mathbf{x}_n - \mu_k)],$$

$$r_{nk} = \rho_{nk} / \sum_j \rho_{nj} \quad \text{(normalised responsibilities)}.$$

M step (update variational parameters for π, μ_k, Λ_k):

$$N_k = \sum_n r_{nk}, \quad \bar{\mathbf{x}}_k = \frac{1}{N_k}\sum_n r_{nk}\mathbf{x}_n, \quad S_k = \frac{1}{N_k}\sum_n r_{nk}(\mathbf{x}_n - \bar{\mathbf{x}}_k)(\mathbf{x}_n - \bar{\mathbf{x}}_k)^\top,$$

$$\alpha_k = \alpha_0 + N_k, \quad \beta_k = \beta_0 + N_k, \quad \mathbf{m}_k = (\beta_0\mathbf{m}_0 + N_k\bar{\mathbf{x}}_k)/\beta_k, \quad \nu_k = \nu_0 + N_k + 1,$$

$$\mathbf{W}_k^{-1} = \mathbf{W}_0^{-1} + N_k S_k + \frac{\beta_0 N_k}{\beta_0 + N_k}(\bar{\mathbf{x}}_k - \mathbf{m}_0)(\bar{\mathbf{x}}_k - \mathbf{m}_0)^\top.$$

Unlike the EM algorithm (Section 5c), VEM marginalises over the parameters, preventing overfitting: components with $N_k \approx 0$ automatically have $\alpha_k \approx \alpha_0$ (returned to prior), effectively pruning unused components. This provides **automatic model order selection**: the number of active components in the VEM solution approximates the correct number of clusters without explicit model comparison.

0.48 5f. Automatic Differentiation Variational Inference

0.48.1 5f.1 The ADVI Framework

Automatic Differentiation Variational Inference (ADVI; Kucukelbir et al., 2017) automates variational inference for any differentiable probabilistic model, making it possible to run VI on complex hierarchical models without manual derivation of the ELBO gradient.

The key insight: transform all latent variables to the real line using differentiable bijections, then apply mean-field or full-rank Gaussian VI in the unconstrained space. For a latent variable θ with support $\mathcal{S}$ (e.g., $\mathcal{S} = \mathbb{R}^+$ for a variance), define a bijection $T : \mathcal{S} \to \mathbb{R}$ (e.g., $T(\sigma) = \log\sigma$ for positive variables). The variational distribution in the transformed space is $q_\phi(\zeta) = \mathcal{N}(\zeta; \mu, \Sigma)$ (mean-field: $\Sigma = \text{diag}(\sigma^2)$, or full-rank: Σ free). The corresponding distribution over $\theta = T^{-1}(\zeta)$ is a transformed Gaussian with density:

$$q(\theta) = q_\phi(T(\theta)) \cdot \left|\det J_T(\theta)\right|.$$

The ELBO in the transformed space:

$$\mathcal{L}(\phi) = \mathbb{E}_{q_\phi(\zeta)}[\log p(x, T^{-1}(\zeta)) + \log\left|\det J_{T^{-1}}(\zeta)\right|] + H(q_\phi),$$

where $H(q_\phi) = \frac{d}{2}(1 + \log 2\pi) + \frac{1}{2}\log\det\Sigma$ is the entropy of the Gaussian q_ϕ.

The gradient $\nabla_\phi\mathcal{L}$ is estimated by the reparametrisation trick: $\zeta = \mu + L\epsilon$, $\epsilon \sim \mathcal{N}(0, I)$ (where L is the Cholesky factor of Σ), and automatic differentiation backpropagates through the probabilistic model and the bijection T^{-1}. ADVI requires no model-specific derivations: any Stan/TensorFlow/JAX-coded model is immediately amenable to VI.

Mean-field vs. full-rank. Mean-field ADVI ($\Sigma = \text{diag}$) has $O(d)$ parameters and is fast but may underestimate posterior correlations. Full-rank ADVI (Σ unconstrained positive definite) has $O(d^2)$ parameters and captures all pairwise correlations, but scales quadratically in d. For models with $d > 1000$, mean-field is typically preferred; low-rank approximations $\Sigma \approx WW^\top + D$ (with $W \in \mathbb{R}^{d\times k}$, $k \ll d$) provide an intermediate option.

Convergence diagnostics. ADVI does not provide $\hat{R}$ or ESS (those are MCMC diagnostics). Instead,

convergence is assessed by: (1) monitoring the ELBO over iterations (convergence when ELBO plateaus); (2) checking the KL divergence between the variational and true posterior (via importance sampling); (3) comparing ADVI mean and variance to MCMC estimates on a subset of the data.

0.49 5e. Variational Autoencoders: Full Mathematical Treatment

0.49.1 5e.1 The VAE Generative Model

The **Variational Autoencoder** (VAE; Kingma and Welling 2014) is a latent variable model with:

- **Prior:** $p(z) = \mathcal{N}(z; 0, I)$ (standard Gaussian in latent space $\mathbb{R}^d$).
- **Decoder:** $p_\theta(x|z) = \mathcal{N}(x; \mu_\theta(z), \sigma_\theta^2(z)I)$ or Bernoulli($g_\theta(z)$) for binary x, where μ_θ and g_θ are neural networks with parameters θ.
- **Encoder (recognition model):** $q_\phi(z|x) = \mathcal{N}(z; \mu_\phi(x), \text{diag}(\sigma_\phi^2(x)))$, a Gaussian with neural-network-parametrised mean and variance.

The objective is the ELBO:

$$\mathcal{L}_{\theta,\phi}(x) = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - \text{KL}(q_\phi(z|x)\|p(z)).$$

The first term is the **reconstruction loss**: how well the decoder reconstructs x from the posterior mean latent code. The second term is the **KL regulariser**: how close the posterior is to the prior. For Gaussian encoder and standard Gaussian prior, the KL has an analytic form:

$$\text{KL}(\mathcal{N}(\mu, \text{diag}(\sigma^2))\|\mathcal{N}(0, I)) = \frac{1}{2}\sum_{j=1}^{d}(\sigma_j^2 + \mu_j^2 - 1 - \log\sigma_j^2).$$

Derivation of the reparametrisation gradient. The gradient $\nabla_\phi \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)]$ cannot be computed by differentiating through the sampling operation $z \sim q_\phi(z|x)$, since the distribution depends on ϕ. The reparametrisation trick replaces this with:

$$z = \mu_\phi(x) + \sigma_\phi(x) \odot \epsilon, \quad \epsilon \sim \mathcal{N}(0, I).$$

Now z is a deterministic function of ϵ and ϕ, and the gradient flows through:

$$\nabla_\phi \mathbb{E}_\epsilon[\log p_\theta(x|\mu_\phi + \sigma_\phi \odot \epsilon)] = \mathbb{E}_\epsilon[\nabla_\phi \log p_\theta(x|\mu_\phi + \sigma_\phi \odot \epsilon)].$$

This expectation is estimated with a single Monte Carlo sample $\epsilon^{(1)} \sim \mathcal{N}(0, I)$ in the standard VAE, giving an unbiased, low-variance gradient estimator.

0.49.2 5e.2 β-VAE and Disentanglement

The β**-VAE** (Higgins et al., 2017) modifies the ELBO by upweighting the KL term:

$$\mathcal{L}^\beta_{\theta,\phi}(x) = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - \beta\ \text{KL}(q_\phi(z|x)\|p(z)),$$

with $\beta > 1$. The increased regularisation pressure forces the encoder to use the minimum number of latent dimensions to represent the data, encouraging **disentanglement**: a latent representation where each dimension z_j encodes an independent generative factor (e.g., one dimension for shape, another for colour, another for orientation).

The information-theoretic interpretation: the β-VAE objective is a constrained rate-distortion problem:

$$\min_{\theta,\phi} -\mathbb{E}[\log p_\theta(x|z)] \quad \text{subject to} \quad \mathbb{E}[\mathrm{KL}(q_\phi(z|x)\|p(z))] \leq C.$$

The Lagrange multiplier for the constraint is $\beta - 1$. As $\beta \to \infty$, the constraint becomes tighter and the encoder is forced to compress more aggressively, prioritising the most informative dimensions of the latent space.

Total Correlation (TC) decomposition. The aggregate KL term decomposes as (Chen et al., 2018):

$$\mathrm{KL}(q_\phi(z|x)\|p(z)) = \underbrace{\mathrm{I}(x;z)}_{\text{mutual information}} + \underbrace{\mathrm{TC}(q_\phi(z))}_{\text{total correlation}} + \underbrace{\mathrm{KL}(q_\phi(z)\|p(z))}_{\text{dimension-wise KL}},$$

where $\mathrm{TC}(q) = \mathrm{KL}(q(z)\|\prod_j q(z_j))$ is the **total correlation**: the KL divergence between the aggregate posterior and the product of its marginals. $\mathrm{TC}(q) = 0$ iff the latent dimensions are marginally independent across the data distribution. The β-VAE primarily penalises the TC term when $\beta > 1$, directly encouraging independence (disentanglement) of the latent dimensions.

0.49.3 5e.3 Hierarchical VAEs and VQ-VAE

Hierarchical VAEs (HVAE; Sønderby et al., 2016; Kingma et al., 2016) extend the VAE with multiple levels of latent variables:

$$p_\theta(x, z_1, \ldots, z_L) = p(z_L) \prod_{l=1}^{L} p_\theta(z_{l-1}|z_l) \cdot p_\theta(x|z_1).$$

The encoder also factorises hierarchically: $q_\phi(z_1, \ldots, z_L|x) = q_\phi(z_L|x) \prod_{l=1}^{L-1} q_\phi(z_l|z_{l+1}, x)$.

The ELBO for an HVAE telescopes:

$$\mathcal{L} = \mathbb{E}[\log p_\theta(x|z_1)] - \sum_{l=1}^{L} \mathbb{E}[\mathrm{KL}(q_\phi(z_l|\cdot)\|p_\theta(z_l|z_{l+1}))],$$

with the KL terms measuring how much each layer's posterior deviates from its prior. Hierarchical VAEs can model highly complex distributions with multiple scales of variation: the top-level latents encode global structure (category, style) while lower-level latents encode local details (texture, fine-grained features).

VQ-VAE (van den Oord et al., 2017) replaces the continuous Gaussian latent space with a discrete codebook $\mathcal{E} = \{e_1, \ldots, e_K\} \subset \mathbb{R}^d$. The encoder maps x to a continuous vector $z_e(x)$, which is then **quantised** to the nearest codebook vector:

$$z_q(x) = e_k \text{ where } k = \arg\min_j \|z_e(x) - e_j\|.$$

The decoder reconstructs from the quantised code. The straight-through estimator copies the gradient from z_q to z_e (since the argmin is not differentiable). VQ-VAE learns a discrete, symbolic latent space, making it particularly suited for modelling categorical variables such as phonemes, words, or action primitives. The codebook is updated by moving each code toward the mean of the encoder outputs assigned to it:

$$\Delta e_k \propto \frac{1}{N_k} \sum_{n:k_n=k} (z_e(x_n) - e_k),$$

which is the k-means update rule: VQ-VAE is equivalent to performing k-means clustering in the latent space simultaneously with training the encoder and decoder.

0.50 6d. Bayesian Deep Learning

0.50.1 6d.1 The Bayesian Neural Network

A **Bayesian Neural Network** (BNN) places a prior distribution over the weights of a neural network and performs Bayesian inference over the posterior $p(w|D)$ given training data D. The predictive distribution for a new input x^*:

$$p(y^*|x^*, D) = \int p(y^*|x^*, w)p(w|D)\, dw.$$

This marginalises over all possible weight configurations, providing calibrated uncertainty: the predictive distribution reflects both the inherent noise in the data and the uncertainty about the model parameters.

Why Bayesian deep learning? Standard deep networks give overconfident predictions on out-of-distribution (OOD) inputs: a network trained on cats and dogs may confidently predict "cat" for an image of a truck. BNNs, by contrast, have high predictive variance on OOD inputs (the posterior is diffuse over weights that make different predictions), providing appropriate uncertainty. This is critical for safety-critical applications (autonomous vehicles, medical diagnosis) where OOD detection is essential.

0.50.2 6d.2 The Laplace Approximation for BNNs

The Laplace approximation for BNNs (MacKay 1992; Ritter et al., 2018): find the MAP weights w_{MAP}, then approximate the posterior as:

$$p(w|D) \approx \mathcal{N}(w; w_{\text{MAP}}, H^{-1}),$$

where $H = -\nabla^2 \log p(w|D)|_{w_{\text{MAP}}}$ is the Hessian of the negative log-posterior at the MAP. The Hessian is the sum of the loss Hessian and the prior precision: $H = H_{\text{loss}} + \lambda I$ (for Gaussian prior with precision λ).

Computing the full Hessian H for a neural network with P parameters is $O(P^2)$ to store and $O(P^3)$ to invert, intractable for large networks. **Kronecker-factored Laplace** (KFLA; Ritter et al., 2018) approximates H as a block-diagonal matrix with Kronecker-factored blocks: $H_l \approx A_l \otimes B_l$ for each layer l, where A_l and B_l are smaller matrices derived from the activation and gradient statistics. The Kronecker structure reduces storage to $O(\sum_l d_{l-1}^2 + d_l^2)$ and inversion to $O(\sum_l d_{l-1}^3 + d_l^3)$.

Last-layer Laplace (Daxberger et al., 2021) further simplifies by applying the Laplace approximation only to the final layer, treating the rest of the network as a deterministic feature extractor. This is computationally cheap ($O(c^2)$ where c is the number of classes) and surprisingly effective: the predictive

uncertainty of last-layer Laplace is competitive with full BNN methods on OOD detection benchmarks.

0.50.3 6d.3 Monte Carlo Dropout as Approximate BNN

MC Dropout (Gal and Ghahramani 2016) is an extremely practical approximation to BNNs: apply dropout at test time and collect T forward passes with different dropout masks. The sample mean is the predictive mean and the sample variance is the predictive uncertainty.

Theoretical justification: a neural network with dropout trained with the standard cross-entropy or MSE loss is equivalent to variational inference with a specific approximate posterior. The variational distribution $q_\phi(w)$ is a product of Bernoulli distributions (each weight is either included or dropped), and the variational objective is the ELBO. The KL divergence term corresponds to the L2 weight decay.

Formally, for a single-layer network $y = f(x, w)$ with Bernoulli dropout (each row of w multiplied by $z_k \sim \text{Bernoulli}(p)$), the effective prior over the network's weights is:

$$q_\phi(w) = \prod_k [p \cdot \mathcal{N}(w_k; m_k, \sigma^2) + (1-p)\delta_0(w_k)],$$

a spike-and-slab distribution with spike probability $1 - p$ (dropout probability). The MC Dropout predictive distribution is an approximation to the posterior predictive of this BNN.

Calibration. A well-calibrated model has predictive confidence matching empirical accuracy: if it says "90% confident," it should be correct 90% of the time. Standard neural networks are often miscalibrated (overconfident). BNNs, Laplace approximations, and temperature scaling are all methods for improving calibration. The **Expected Calibration Error** (ECE) measures miscalibration: $\text{ECE} = \sum_b \frac{|B_b|}{n} |\text{acc}(B_b) - \text{conf}(B_b)|$ where B_b are confidence bins.

0.51 6c. Gaussian Process Regression: Complete Worked Example

0.51.1 6c.1 A Concrete Regression Problem

Consider the task of learning a smooth nonlinear function $f : [0, 10] \to \mathbb{R}$ from noisy observations. We observe $n = 10$ data points (x_i, y_i) where $y_i = f(x_i) + \epsilon_i$, $\epsilon_i \sim \mathcal{N}(0, \sigma_n^2)$. We want to:

1. Infer the posterior distribution over f given the data.
2. Make predictions with calibrated uncertainty for new test inputs.
3. Automatically select the kernel hyperparameters from the data.

Model specification. Prior: $f \sim \mathcal{GP}(0, k_\theta)$ with squared exponential kernel $k_\theta(x, x') = \sigma_f^2 \exp(-|x - x'|^2/2\ell^2)$ and noise σ_n^2. Hyperparameters: $\theta = (\sigma_f^2, \ell, \sigma_n^2)$.

Joint distribution. The joint distribution of the training outputs $\mathbf{y}$ and the function value f_* at a test point x_* is:

$$\begin{pmatrix} \mathbf{y} \\ f_* \end{pmatrix} \sim \mathcal{N}\left(\mathbf{0},\ \begin{pmatrix} K_{XX} + \sigma_n^2 I & \mathbf{k}_* \\ \mathbf{k}_*^\top & k_{**} \end{pmatrix}\right),$$

where K_{XX} is the $n \times n$ training kernel matrix, $\mathbf{k}_* = (k_\theta(x_*, x_i))_{i=1}^n$ is the cross-covariance vector, and $k_{**} = k_\theta(x_*, x_*)$ is the test-point variance.

Posterior predictive. Conditioning on $\mathbf{y}$ (Section 7.1):

$$f_*|X, \mathbf{y}, x_* \sim \mathcal{N}(\bar{f}_*, \mathbb{V}[f_*]),$$

$$\bar{f}_* = \mathbf{k}_*^\top (K_{XX} + \sigma_n^2 I)^{-1}\mathbf{y}, \qquad \mathbb{V}[f_*] = k_{**} - \mathbf{k}_*^\top (K_{XX} + \sigma_n^2 I)^{-1}\mathbf{k}_*.$$

The posterior mean $\bar{f}_*$ is a weighted sum of the training outputs with kernel-based weights. The posterior variance $\mathbb{V}[f_*]$ equals the prior variance k_{**} minus the **information reduction** from the training data. At training points, $\mathbb{V}[f(x_i)] \approx \sigma_n^2$ (residual measurement uncertainty); far from training points, $\mathbb{V}[f_*] \to \sigma_f^2$ (prior uncertainty).

0.51.2 6c.2 Hyperparameter Learning via Marginal Likelihood

The log marginal likelihood as a function of $\theta = (\sigma_f^2, \ell, \sigma_n^2)$:

$$\log p(\mathbf{y}|X, \theta) = -\frac{1}{2}\mathbf{y}^\top K_y^{-1}\mathbf{y} - \frac{1}{2}\log\det K_y - \frac{n}{2}\log 2\pi,$$

where $K_y = K_{XX} + \sigma_n^2 I$. The three terms have interpretations:

- $-\frac{1}{2}\mathbf{y}^\top K_y^{-1}\mathbf{y}$: **data fit** — rewards models that assign high likelihood to the observed data.
- $-\frac{1}{2}\log\det K_y$: **complexity penalty** — penalises models with large prior variance or small noise (which can fit any data too easily).
- $-\frac{n}{2}\log 2\pi$: normalisation constant.

Automatic relevance determination (ARD) uses a separate length-scale per dimension: $k(x, x') = \sigma_f^2 \exp(-\frac{1}{2}\sum_d (x_d - x'_d)^2/\ell_d^2)$. Irrelevant input dimensions get large ℓ_d, effectively removing them from the kernel. The marginal likelihood selects the right length-scales automatically.

0.51.3 6c.3 Numerical Example

Suppose the true function is $f(x) = \sin(x) + 0.5\sin(2x)$ and we observe $n = 10$ points with $\sigma_n = 0.3$. The MLE hyperparameters are approximately $\ell^* \approx 1.8$ (captures the dominant period $\approx 2\pi$ of the function), $\sigma_f^* \approx 1.2$ (matches the amplitude), $\sigma_n^* \approx 0.31$ (close to the true noise).

The posterior predictive at $x_* = 7.5$ (not a training point): $\bar{f}(7.5) \approx f(7.5) \pm 0.35$ (one standard deviation). At $x_* = 12$ (extrapolation beyond training range): $\bar{f}(12) \approx 0$ (returns to prior mean), $\mathbb{V}[f(12)] \approx \sigma_f^{*2}$ (returns to prior variance). The GP correctly reflects the increased uncertainty in extrapolation.

0.52 6b. Dirichlet Processes and Bayesian Nonparametrics

0.52.1 6b.1 The Dirichlet Process as a Distribution over Distributions

A **Dirichlet Process** $G \sim \text{DP}(\alpha, G_0)$ is a random probability measure on a measurable space $(\Theta, \mathcal{B})$. It is characterised by: for any finite measurable partition $(A_1, \ldots, A_k)$ of Θ:

$$(G(A_1), \ldots, G(A_k)) \sim \text{Dirichlet}(\alpha G_0(A_1), \ldots, \alpha G_0(A_k)).$$

The concentration parameter $\alpha > 0$ controls how closely G resembles the base measure G_0: as $\alpha \to \infty$, $G \to G_0$ almost surely; as $\alpha \to 0$, G concentrates on a single point drawn from G_0.

Almost surely, draws from a DP are **discrete**: $G = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k}$ where $\theta_k \overset{\text{i.i.d.}}{\sim} G_0$ and $\pi = (\pi_1, \pi_2, \ldots)$ follows the GEM distribution with parameter α (the **stick-breaking** construction):

$$v_k \overset{\text{i.i.d.}}{\sim} \text{Beta}(1, \alpha), \quad \pi_k = v_k \prod_{j=1}^{k-1} (1 - v_j).$$

The atoms θ_k and weights π_k are independent; the weights sum to 1 almost surely.

Posterior conjugacy. Given i.i.d. observations $\theta_1, \ldots, \theta_n \sim G$ and $G \sim \text{DP}(\alpha, G_0)$, the posterior is:

$$G | \theta_{1:n} \sim \text{DP}\left(\alpha + n,\ \frac{\alpha}{\alpha + n} G_0 + \frac{1}{\alpha + n} \sum_{i=1}^{n} \delta_{\theta_i} \right).$$

The posterior base measure is a mixture of the prior G_0 and the empirical measure $\frac{1}{n} \sum_i \delta_{\theta_i}$, with weights proportional to α and n respectively. New draws from the posterior base measure come from G_0 with probability $\alpha/(\alpha + n)$ and equal each observed θ_i with probability $1/(\alpha + n)$: this is the **Pólya urn** representation.

0.52.2 6b.2 The Chinese Restaurant Process and Clustering

The **Chinese Restaurant Process** (CRP) characterises the marginal distribution over cluster assignments when the cluster parameters are integrated out. Imagine n customers entering a restaurant with infinitely many tables:

- Customer 1 sits at table 1.
- Customer $n+1$ sits at an **occupied** table k (with n_k current customers) with probability $n_k/(\alpha+n)$, or starts a **new** table with probability $\alpha/(\alpha + n)$.

The resulting partition ρ_n of $\{1, \ldots, n\}$ into tables is exchangeable: its probability depends only on the sizes of the tables, not on which customers sit at which table. The number of tables $K_n \sim \alpha \log n$ (Ewens 1972): the DP model automatically infers a number of clusters that grows logarithmically with the data.

Collapsed Gibbs sampler for DP mixtures. Integrating out G, the posterior over cluster assignments is sampled by iterating: for each observation i, sample:

$$z_i | z_{-i}, x_{1:n} \sim \sum_{k=1}^{K_{-i}} \frac{n_{-i,k}}{\alpha + n - 1} p(x_i | \phi_k, z_{-i}) \cdot \delta_k + \frac{\alpha}{\alpha + n - 1} \int p(x_i | \phi) G_0(d\phi) \cdot \delta_{\text{new}},$$

where $n_{-i,k}$ is the number of observations (excluding i) in cluster k, K_{-i} is the current number of clusters, and ϕ_k is the cluster parameter estimated from the cluster members excluding i. The second term creates a new cluster with the prior probability of x_i under the base measure.

0.52.3 6b.3 Gaussian Processes as Dirichlet Process Limits

There is a deep connection between GPs and DPs: both are **functional** priors, placing distributions over functions (GP) or measures (DP). The GP is the appropriate nonparametric prior for regression (contin-

uous function estimation); the DP is appropriate for density estimation (discrete mixture modelling).

The **DP mixture of Gaussians** (DPMG) is a flexible density estimator: $f(x) = \int \mathcal{N}(x; \mu, \sigma^2) G(d\mu\, d\sigma^2)$ where $G \sim \text{DP}(\alpha, G_0)$. As the number of observations grows, the DPMG consistently estimates any continuous density (Ghosal and van der Vaart 2017), making it the nonparametric Bayesian analogue of kernel density estimation. The DP provides automatic bandwidth selection: the concentration parameter α controls the effective bandwidth, and its posterior is updated from the data.

For cognitive science, DP mixture models are natural priors for categorical perception: perceived categories emerge from the data without pre-specifying the number of categories. The CRP prior assigns higher probability to fewer, larger clusters (because the rich-get-richer dynamics make existing tables attractive), implementing a form of Occam's razor: the model prefers simpler (fewer categories) explanations unless the data strongly support additional categories.

0.53 5d. Stein Variational Gradient Descent

0.53.1 5d.1 Kernelised Stein Operators

Stein Variational Gradient Descent (SVGD; Liu and Wang 2016) is a particle-based variational inference method that simultaneously optimises a set of M particles $\{x_i\}_{i=1}^M$ to approximate the posterior p, using gradient information.

The key idea: find a perturbation $\phi^* : \mathcal{X} \to \mathcal{X}$ (a velocity field) that maximally decreases the KL divergence $\text{KL}(q\|p)$ when the particles are updated as $x \leftarrow x + \epsilon\phi^*(x)$. The update decreases KL at rate:

$$\frac{d}{d\epsilon} \text{KL}(q_{[\epsilon\phi]}\|p)\bigg|_{\epsilon=0} = -\mathbb{E}_{x\sim q}[\nabla \cdot \phi(x) + \phi(x)^\top \nabla \log p(x)].$$

Define the **Stein operator** $\mathcal{A}_p\phi(x) = \phi(x)^\top \nabla \log p(x) + \nabla \cdot \phi(x)$. The steepest descent direction in the unit ball of the RKHS $\mathcal{H}$ is:

$$\phi^*(x) = \mathbb{E}_{x'\sim q}[\nabla_{x'} \log p(x') k(x', x) + \nabla_{x'} k(x', x)],$$

where k is the reproducing kernel. The two terms have interpretations: the first drives particles toward high-probability regions (gradient of log-target), and the second creates a "repulsion" force that prevents particles from collapsing to a single point (the repulsion is mediated by the kernel, strongest between nearby particles).

SVGD algorithm. Initialise M particles $\{x_i^{(0)}\} \sim q_0$. At each step:

$$x_i^{(t+1)} = x_i^{(t)} + \epsilon_t \hat{\phi}^*(x_i^{(t)}),$$

$$\hat{\phi}^*(x_i) = \frac{1}{M} \sum_{j=1}^{M} \left[k(x_j, x_i) \nabla_{x_j} \log p(x_j) + \nabla_{x_j} k(x_j, x_i)\right].$$

The cost per step is $O(M^2 d)$ (summing over all M^2 particle pairs in d dimensions). SVGD converges to the target distribution as $M \to \infty$ and step size $\epsilon_t \to 0$ appropriately.

SVGD has a compelling information-geometric interpretation: it is gradient flow on the Wasserstein-2

space of measures, driven by the Stein operator. The particles' empirical distribution follows the gradient of $\mathrm{KL}(q\|p)$ in the metric of the RKHS, making SVGD a "kernelised Wasserstein gradient flow."

For cognitive modelling, SVGD provides a biologically plausible inference algorithm: M particles can be interpreted as a population of neurons, each encoding a hypothesis about the current state, and the SVGD update rule is local (each particle sees only the other particles through the kernel, analogous to lateral connections in a neural population).

0.53.2 5d.2 Black-Box Variational Inference

Black-Box VI (Ranganath et al., 2014) reduces variational inference to stochastic gradient ascent on the ELBO, requiring only the ability to evaluate $\log p(x, z)$ (not gradients of the model) and to sample from $q_\lambda(z)$ (and evaluate its log-density).

The score function estimator (REINFORCE-based) gives:

$$\nabla_\lambda \mathcal{L} = \mathbb{E}_{q_\lambda(z)}\left[\nabla_\lambda \log q_\lambda(z) \cdot \log \frac{p(x,z)}{q_\lambda(z)}\right].$$

This estimator is unbiased but has high variance. **Variance reduction** via control variates: replacing $\log(p/q)$ by $\log(p/q) - b$ (where b is a baseline chosen to minimise variance) gives an estimator with the same expectation but lower variance. The optimal constant baseline is $b^* = \mathbb{E}[\nabla_\lambda \log q \cdot \log(p/q)]/\mathbb{E}[\nabla_\lambda \log q]$.

The **reparametrisation trick** (Section 5.4) gives a lower-variance estimator when z can be reparametrised as $z = g(\epsilon, \lambda)$ with $\epsilon \sim p(\epsilon)$ independent of λ:

$$\nabla_\lambda \mathcal{L} = \mathbb{E}_{p(\epsilon)}[\nabla_\lambda \log p(x, g(\epsilon,\lambda)) - \nabla_\lambda \log q_\lambda(g(\epsilon,\lambda))].$$

The variance is typically much lower than the score estimator because the gradient flows through g directly. For distributions with no natural reparametrisation (Gamma, Beta, Dirichlet), the implicit reparametrisation gradient (Figurnov et al., 2018) provides a differentiable sampler through the inverse CDF.

0.54 5b. Advanced Variational Methods

0.54.1 5b.1 The Importance-Weighted Autoencoder

The VAE's ELBO is a lower bound on the log marginal likelihood. The **Importance-Weighted Autoencoder** (IWAE; Burda et al., 2015) tightens this bound using multiple samples:

$$\mathcal{L}_K(\theta, \phi; x) = \mathbb{E}_{z_1,\ldots,z_K \sim q_\phi(z|x)}\left[\log \frac{1}{K}\sum_{k=1}^{K} \frac{p_\theta(x, z_k)}{q_\phi(z_k|x)}\right] \leq \log p_\theta(x).$$

The bound $\mathcal{L}_K$ tightens as K increases: $\mathcal{L}_1 = \text{ELBO} \leq \mathcal{L}_2 \leq \cdots \leq \mathcal{L}_\infty = \log p_\theta(x)$.

Proof of monotonicity. By Jensen's inequality applied to the concave log function and the convex com-

bination of importance weights:

$$\mathcal{L}_{K+1} - \mathcal{L}_K = \mathbb{E}\left[\log \frac{1}{K+1}\sum_{k=1}^{K+1} w_k - \log \frac{1}{K}\sum_{k=1}^{K} w_k\right] \geq 0,$$

since the left expectation is a log-ratio of a $(K+1)$-sample estimate to a K-sample estimate of the same quantity, and adding samples to an importance sampling estimate improves it in expectation. □

Gradient computation for $\mathcal{L}_K$ uses the reparametrisation trick on each z_k: $z_k = \mu_\phi(x) + \sigma_\phi(x) \odot \epsilon_k$, $\epsilon_k \sim \mathcal{N}(0, I)$, and the gradient of the log-sum-exp is:

$$\nabla_\phi \mathcal{L}_K = \mathbb{E}\left[\sum_k \tilde{w}_k \nabla_\phi \log w_k\right], \qquad \tilde{w}_k = \frac{w_k}{\sum_j w_j}.$$

The normalised importance weights $\tilde{w}_k$ act as a self-normalised estimator; the gradients are weighted by these normalised weights, emphasising samples with high importance weight.

0.54.2 5b.2 Stochastic Variational Inference

For large datasets (millions of observations), computing the ELBO requires summing over all data points, making full-batch gradient ascent prohibitive. **Stochastic variational inference** (SVI; Hoffman et al., 2013) uses **stochastic gradient ascent** on the ELBO using minibatches.

For a model with N observations and local latent variables z_n (one per observation) and global parameters β, the ELBO factorises:

$$\mathcal{L} = \mathbb{E}_q[\log p(\beta)] - \mathrm{KL}(q(\beta)\|p(\beta)) + \sum_{n=1}^{N} \mathbb{E}_q[\log p(x_n, z_n|\beta)] - \mathbb{E}_q[\log q(z_n)].$$

A minibatch estimate uses a random subset $\mathcal{S} \subseteq \{1, \ldots, N\}$ of size S:

$$\hat{\mathcal{L}}_{\mathcal{S}} = \mathbb{E}_q[\log p(\beta)] - \mathrm{KL}(q(\beta)\|p(\beta)) + \frac{N}{S}\sum_{n\in\mathcal{S}} \mathbb{E}_q[\log p(x_n, z_n|\beta) - \log q(z_n)].$$

The scaling factor N/S corrects for the smaller minibatch size, yielding an unbiased estimate of $\mathcal{L}$.

The natural gradient (the gradient pre-multiplied by $\mathcal{I}^{-1}$) improves convergence for exponential family variational parameters. For mean-field distributions in exponential form, the natural gradient coordinate ascent update for the global variational parameter λ reduces to:

$$\lambda^{(t+1)} = (1 - \rho_t)\lambda^{(t)} + \rho_t \hat{\lambda}^*,$$

where $\hat{\lambda}^*$ is the optimal local update for the current minibatch and ρ_t is a learning rate satisfying $\sum_t \rho_t = \infty$, $\sum_t \rho_t^2 < \infty$ (Robbins-Monro conditions). This interpolation between current and optimal local parameters is both stable and efficient.

0.54.3 5b.3 Normalising Flows: Proofs and Properties

For a normalising flow $z_K = f_K \circ \cdots \circ f_1(z_0)$ with $z_0 \sim q_0$, the change-of-variables formula for densities gives:

$$\log q_K(z_K) = \log q_0(z_0) - \sum_{k=1}^{K} \log \left| \det J_{f_k}(z_{k-1}) \right| .$$

Proof. By the substitution rule for densities: if $z = f(u)$ with f invertible and differentiable, $q_z(z)|dz| = q_u(u)|du|$, so $q_z(f(u)) = q_u(u)/|\det J_f(u)|$. Apply this inductively for each transformation. □

The computational challenge is the Jacobian determinant. For **affine coupling layers** (Dinh et al., 2014, RealNVP), the transformation partitions the input $z = (z_A, z_B)$ and applies:

$$z'_A = z_A, \qquad z'_B = z_B \odot \exp(s(z_A)) + t(z_A),$$

where $s, t : \mathbb{R}^{d/2} \to \mathbb{R}^{d/2}$ are arbitrary (possibly deep) neural networks. The Jacobian is triangular with unit diagonal block for z_A and diagonal block $\operatorname{diag}(\exp(s(z_A)))$ for z_B:

$$\det J = \prod_i \exp(s(z_A)_i) = \exp(\mathbf{1}^\top s(z_A)),$$

computable in $O(d)$ rather than $O(d^3)$. The inverse $z_B = (z'_B - t(z_A)) \odot \exp(-s(z_A))$ is also cheap, making both sampling ($z_0 \to z_K$) and density evaluation ($z_K \to z_0$) efficient.

Autoregressive flows (MAF, IAF; Papamakarios et al., 2017) use transformations where each component z_k depends only on $z_{1:k-1}$:

$$z'_k = h(z_k; \mu_k(z_{1:k-1}), \sigma_k(z_{1:k-1})),$$

with h an invertible elementwise function. The Jacobian is lower triangular, so its determinant is the product of diagonal elements. Masked Autoregressive Flow (MAF) computes the forward pass in $O(d)$ but requires $O(d)$ sequential steps for the inverse; Inverse Autoregressive Flow (IAF) swaps these, enabling fast sampling.

Section *III*

Section III — Gaussian Processes and Probabilistic Function Spaces

0.55 6. Functions as Random Variables

In the preceding Section s, we placed probability distributions over finite-dimensional parameter vectors $\theta \in \mathbb{R}^d$. But in many cognitive problems, we wish to reason directly about **functions**: given a set of input-output observations, what function $f : \mathcal{X} \to \mathbb{R}$ is consistent with those observations, and how uncertain should we be about f's values at unobserved points?

This requires probability distributions over function spaces: infinite-dimensional objects. The mathematical machinery for this is **Gaussian process** (GP) theory.

Definition 0.55.1 (Gaussian Process)**.** A Gaussian process is a collection of random variables $\{f(x) : x \in \mathcal{X}\}$, indexed by an input space $\mathcal{X}$, such that for any finite set of inputs $\{x_1, \ldots, x_n\} \subset \mathcal{X}$, the joint distribution of $(f(x_1), \ldots, f(x_n))$ is multivariate Gaussian.

A GP is fully specified by its **mean function** $m : \mathcal{X} \to \mathbb{R}$ and its **covariance function** (or **kernel**) $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$:

$$m(x) = \mathbb{E}[f(x)], \quad k(x, x') = \text{Cov}(f(x), f(x')) = \mathbb{E}[(f(x) - m(x))(f(x') - m(x'))].$$

We write $f \sim \mathcal{GP}(m, k)$. For any finite index set $X = (x_1, \ldots, x_n)$, the marginal is $(f(x_1), \ldots, f(x_n)) \sim \mathcal{N}(\mathbf{m}, \mathbf{K})$ where $m_i = m(x_i)$ and $K_{ij} = k(x_i, x_j)$.

0.56 7. Kernels and the Geometry of Beliefs

0.56.1 7.1 Positive Semi-Definiteness

A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a valid covariance function — a valid kernel — if and only if it is **symmetric** ($k(x, x') = k(x', x)$) and **positive semi-definite**: for all n, all $x_1, \ldots, x_n \in \mathcal{X}$, and all $c_1, \ldots, c_n \in \mathbb{R}$,

$$\sum_{i=1}^{n} \sum_{j=1}^{n} c_i c_j k(x_i, x_j) \geq 0.$$

Equivalently, the $n \times n$ Gram matrix K with $K_{ij} = k(x_i, x_j)$ must be positive semi-definite for all finite index sets.

The positive semi-definiteness condition is necessary because covariance matrices must be positive semi-definite: for any random vector $(f(x_1), \ldots, f(x_n))$, the variance of any linear combination $\sum_i c_i f(x_i)$ is $\sum_{i,j} c_i c_j \, \text{Cov}(f(x_i), f(x_j)) \geq 0$. A kernel encodes a prior belief about the smoothness, periodicity, or compositionality of the unknown function.

0.56.2 7.2 Standard Kernels

The **squared exponential** (SE) kernel is the most commonly used GP kernel:

$$k_{\text{SE}}(x, x') = \sigma^2 \exp\left(-\frac{\|x - x'\|^2}{2\ell^2} \right).$$

The hyperparameter σ^2 is the output variance (the prior variance at any single point) and ℓ is the **length-scale** controlling how quickly the function varies. Functions drawn from a GP with the SE kernel are

infinitely differentiable: the SE kernel corresponds to an extremely smooth prior. In terms of Fourier analysis (Bochner's theorem), the SE kernel corresponds to a Gaussian spectral density, meaning the process has power at all frequencies but with exponentially decreasing amplitude.

The **Matérn** kernel family provides a more flexible smoothness parameter:

$$k_\nu(x, x') = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}\|x - x'\|}{\ell} \right)^\nu K_\nu \left(\frac{\sqrt{2\nu}\|x - x'\|}{\ell} \right),$$

where K_ν is the modified Bessel function of the second kind and $\nu > 0$ controls smoothness. A GP with $\nu = p + 1/2$ (for non-negative integer p) is exactly p times mean-square differentiable. The special cases $\nu = 1/2$ (Ornstein-Uhlenbeck, exponential kernel, once differentiable), $\nu = 3/2$, and $\nu = 5/2$ are widely used in practice.

The **periodic kernel** $k_{\text{per}}(x, x') = \sigma^2 \exp(-2\sin^2(\pi|x - x'|/T)/\ell^2)$ encodes periodicity with period T. It is obtained by mapping inputs through the feature map $\phi(x) = [\sin(2\pi x/T), \cos(2\pi x/T)]$ and applying the SE kernel, a standard construction for deriving kernels for structured domains.

Kernels can be combined: sums $k_1 + k_2$ and products $k_1 \cdot k_2$ of valid kernels are valid kernels. This compositionality, analogous to combining features in a model, is the formal basis for the *Automatic Statistician* programme of automatically discovering the structure of time series data by searching over kernel combinations.

0.56.3 7.3 GP Regression: Exact Posterior Inference

Suppose we observe n noisy function evaluations $\mathbf{y} = (y_1, \ldots, y_n)^\top$ at inputs $X = (x_1, \ldots, x_n)$, where $y_i = f(x_i) + \epsilon_i$ and $\epsilon_i \sim \mathcal{N}(0, \sigma_n^2)$ are independent noise terms. We place a GP prior $f \sim \mathcal{GP}(0, k)$ (zero mean function for simplicity) and wish to compute the posterior distribution of f at test inputs $X_* = (x_*^1, \ldots, x_*^m)$.

The joint distribution of $(\mathbf{y}, \mathbf{f}_*)$ (where $\mathbf{f}_* = (f(x_*^1), \ldots, f(x_*^m))$) under the prior is:

$$\begin{pmatrix} \mathbf{y} \\ \mathbf{f}_* \end{pmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{pmatrix} K_{XX} + \sigma_n^2 I & K_{XX_*} \\ K_{X_*X} & K_{X_*X_*} \end{pmatrix} \right),$$

where $[K_{XX}]_{ij} = k(x_i, x_j)$, $[K_{XX_*}]_{ij} = k(x_i, x_*^j)$, and so on. Using the formula for the conditional distribution of a jointly Gaussian vector, the posterior is:

$$\mathbf{f}_* \mid X, \mathbf{y}, X_* \sim \mathcal{N}(\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_*),$$

$$\boldsymbol{\mu}_* = K_{X_*X}(K_{XX} + \sigma_n^2 I)^{-1}\mathbf{y},$$

$$\boldsymbol{\Sigma}_* = K_{X_*X_*} - K_{X_*X}(K_{XX} + \sigma_n^2 I)^{-1}K_{XX_*}.$$

This is **exact closed-form Bayesian inference in a function space**. The posterior mean $\boldsymbol{\mu}_*$ is a linear smoother of the observations $\mathbf{y}$, with weights given by the kernel similarities between test and training points. The posterior variance $\boldsymbol{\Sigma}_*$ quantifies uncertainty: it is lower in regions of high observation density and higher far from observations. No approximation has been made.

This tractability is specific to Gaussian likelihoods. For non-Gaussian likelihoods (e.g., Bernoulli outputs

in classification), the posterior is non-Gaussian and approximate inference methods (Laplace approximation, expectation propagation, or variational inference) are required.

In the cognitive context, GP regression is a model of how an agent infers a continuous hidden function (a reward landscape, a sensory field, a tuning curve) from sparse noisy observations. The GP posterior is the agent's complete probabilistic belief about the function's values at all unobserved points.

0.57 7b. Sparse Gaussian Processes

0.57.1 7b.1 The Computational Bottleneck

Exact GP regression requires inverting the $n \times n$ kernel matrix $K_{XX} + \sigma_n^2 I$, at cost $O(n^3)$ in time and $O(n^2)$ in memory. For $n \gtrsim 10^4$ observations this is prohibitive. **Sparse GP** methods introduce a small set of $m \ll n$ **inducing points** $Z = (z_1, \ldots, z_m) \subset \mathcal{X}$ and approximate the posterior using only the function values $\mathbf{u} = (f(z_1), \ldots, f(z_m))$ at these points.

The **Fully Independent Training Conditional** (FITC) approximation (Snelson and Ghahramani, 2006) replaces the exact covariance with a low-rank plus diagonal structure: the joint covariance of $(f(X), f(X_*))$ is approximated as:

$$\tilde{K} = Q_{**} + \text{diag}(K_{XX} - Q_{XX}),$$

where $Q_{AB} = K_{AZ} K_{ZZ}^{-1} K_{ZB}$ is the projected covariance through the inducing points. This retains the correct marginal variances (the diagonal correction $K_{XX} - Q_{XX}$) while using only rank-m off-diagonal structure.

The more principled **variational sparse GP** (Titsias, 2009) treats the inducing variables $\mathbf{u}$ as variational parameters. The variational distribution is $q(f) = p(f|\mathbf{u})q(\mathbf{u})$ where $p(f|\mathbf{u})$ is the exact GP conditional and $q(\mathbf{u}) = \mathcal{N}(\mathbf{m}, S)$ is optimised. The ELBO becomes:

$$\mathcal{L} = \sum_{i=1}^{n} \mathbb{E}_{q(f_i)}[\log p(y_i|f_i)] - \text{KL}(q(\mathbf{u})\|p(\mathbf{u})),$$

where $q(f_i) = \mathcal{N}(f_i; \mu_i, \sigma_i^2)$ is the marginal of the approximate posterior at x_i:

$$\mu_i = K_{x_i Z} K_{ZZ}^{-1} \mathbf{m}, \quad \sigma_i^2 = k(x_i, x_i) - K_{x_i Z} K_{ZZ}^{-1} (K_{ZZ} - S) K_{ZZ}^{-1} K_{Z x_i}.$$

Optimising over $\mathbf{m}$, S, and Z simultaneously costs $O(nm^2 + m^3)$ per gradient step, reducing cubic to linear scaling in n for fixed m.

0.57.2 7b.2 GP Classification

For binary classification, the likelihood $p(y_i = 1|f(x_i)) = \sigma(f(x_i))$ (where σ is the logistic function) is non-Gaussian, and the GP posterior is no longer available in closed form. The two standard approximations are:

Laplace approximation. Find the posterior mode $\hat{f}$ by Newton's method on $\log p(f|y) = \log p(y|f) -$

$\frac{1}{2} f^\top K^{-1} f + \text{const}$, then approximate the posterior as Gaussian around $\hat{f}$:

$$\hat{f} = \arg\max_f \log p(y|f) - \frac{1}{2} f^\top K^{-1} f,$$

$$p(f|y) \approx \mathcal{N}(f; \hat{f}, (K^{-1} + W)^{-1}),$$

where $W = -\nabla^2 \log p(y|f)|_{f=\hat{f}}$ is the diagonal matrix of likelihood curvatures (for logistic likelihood, $W_{ii} = \pi_i(1-\pi_i)$ where $\pi_i = \sigma(\hat{f}(x_i))$). The Woodbury identity $K - K(K + W^{-1})^{-1}K = (K^{-1} + W)^{-1}$ allows computation in $O(n^3)$.

Expectation propagation (EP). EP approximates the non-Gaussian likelihood factors $t_i(f_i) = p(y_i|f_i)$ by Gaussian *site approximations* $\tilde{t}_i(f_i) = \tilde{Z}_i \mathcal{N}(f_i; \tilde{\mu}_i, \tilde{\sigma}_i^2)$ and iteratively refines them by matching moments. EP is typically more accurate than Laplace for classification and provides a better approximation to the marginal likelihood.

0.57.3 7b.3 The Neural Network Gaussian Process

A striking connection between GPs and neural networks was established by Neal (1996) and extended by Jacot et al. (2018). Consider a one-hidden-layer network:

$$f(x) = \frac{1}{\sqrt{H}} \sum_{h=1}^{H} v_h \phi(w_h^\top x + b_h),$$

with $w_h \sim \mathcal{N}(0, \sigma_w^2 I)$, $b_h \sim \mathcal{N}(0, \sigma_b^2)$, $v_h \sim \mathcal{N}(0, 1)$. By the central limit theorem, as $H \to \infty$, the function f converges in distribution to a GP:

$$f \xrightarrow{d} \mathcal{GP}(0, k_{\text{NNGP}}),$$

where the NNGP kernel is $k_{\text{NNGP}}(x, x') = \mathbb{E}_{w,b}[\phi(w^\top x + b)\phi(w^\top x' + b)]$. Deep networks with L hidden layers yield deep kernel recursions: $K^{(l)}(x, x') = \mathbb{E}_{f \sim \mathcal{GP}(0, K^{(l-1)})}[\phi(f(x))\phi(f(x'))]$.

This **neural network–GP correspondence** has profound implications: in the infinite-width limit, a Bayesian neural network with Gaussian weight priors is equivalent to a GP regression with a specific kernel. The kernel encodes the inductive bias of the architecture (depth, activation function, skip connections) in a mathematically precise way. Conversely, finite-width corrections, captured by the Neural Tangent Kernel (NTK) dynamics, govern how finite networks deviate from the GP limit during gradient-based training.

0.58 7c. Spectral Theory of Kernels

0.58.1 7c.1 Mercer's Theorem

Every symmetric positive semi-definite kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ on a compact domain $\mathcal{X}$ has a **spectral decomposition**:

Theorem 0.58.1 (Mercer 1909)**.** Let $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a continuous, symmetric, positive semi-definite

kernel on a compact metric space $\mathcal{X}$. Then:

$$k(x, x') = \sum_{j=1}^{\infty} \lambda_j \phi_j(x) \phi_j(x'),$$

where $(\lambda_j, \phi_j)_{j=1}^{\infty}$ are the eigenvalue-eigenfunction pairs of the integral operator $T_k : L^2(\mathcal{X}) \to L^2(\mathcal{X})$, $(T_k f)(x) = \int k(x, x') f(x')\, dx'$. The eigenvalues $\lambda_j \geq 0$ and the eigenfunctions ϕ_j are orthonormal in $L^2(\mathcal{X})$. The sum converges absolutely and uniformly.

Proof sketch. T_k is a compact self-adjoint operator on $L^2(\mathcal{X})$ (by positivity and continuity of k). By the spectral theorem for compact self-adjoint operators, it has a complete orthonormal basis of eigenfunctions $\{\phi_j\}$ with non-negative eigenvalues $\lambda_j \to 0$. The kernel's representation follows from $k(x, x') = \langle \delta_x, T_k \delta_{x'} \rangle$ and the eigenfunction expansion. □

The Mercer representation has three consequences for GP theory:

Feature map. Define $\Phi(x) = (\sqrt{\lambda_1}\phi_1(x), \sqrt{\lambda_2}\phi_2(x), \ldots) \in \ell^2$. Then $k(x, x') = \langle \Phi(x), \Phi(x') \rangle_{\ell^2}$: every kernel is an inner product in an (infinite-dimensional) feature space. The GP with kernel k is equivalent to a Bayesian linear model in this infinite-dimensional feature space.

RKHS. The **reproducing kernel Hilbert space** (RKHS) $\mathcal{H}_k$ associated with k is the closure of $\{f : f(x) = \sum_j a_j k(x, x_j)\}$ with inner product $\langle k(\cdot, x), k(\cdot, x') \rangle_{\mathcal{H}_k} = k(x, x')$. The **reproducing property** is $\langle f, k(\cdot, x) \rangle_{\mathcal{H}_k} = f(x)$ for all $f \in \mathcal{H}_k$. The norm $\|f\|_{\mathcal{H}_k}^2 = \sum_j a_j^2 / \lambda_j$ penalises functions that place mass on directions with small eigenvalues (high-frequency directions), encoding a smoothness prior.

GP sample paths. Functions drawn from a GP with kernel k lie (almost surely) in $\mathcal{H}_k$ only when $\sum_j \lambda_j < \infty$ (trace-class kernel). For the SE kernel, eigenvalues decay faster than any polynomial, and samples are almost surely smooth. For the Matérn-ν kernel on $\mathbb{R}^d$, eigenvalues decay as $\lambda_j \sim j^{-(2\nu+d)/d}$, and samples are ν-times mean-square differentiable, consistent with Sobolev space theory.

0.58.2 7c.2 Bochner's Theorem and Spectral Densities

For translation-invariant kernels $k(x, x') = k(x - x')$ on $\mathbb{R}^d$, Bochner's theorem characterises valid kernels in terms of their Fourier transforms:

Theorem 0.58.2 (Bochner 1932)**.** A continuous, translation-invariant function $k : \mathbb{R}^d \to \mathbb{R}$ is a positive semidefinite kernel iff it is the Fourier transform of a non-negative finite measure Λ on $\mathbb{R}^d$:

$$k(x - x') = \int_{\mathbb{R}^d} e^{i\omega^\top (x - x')}\, d\Lambda(\omega).$$

If Λ has density $S(\omega)$ (the **spectral density**), then $k(\tau) = \int S(\omega) e^{i\omega^\top \tau}\, d\omega$.

For the squared exponential kernel: $k_{\text{SE}}(\tau) = \sigma^2 \exp(-\|\tau\|^2 / 2\ell^2)$ has spectral density $S(\omega) = \sigma^2 (2\pi\ell^2)^{d/2} \exp(-\ell^2 \|\omega\|^2$ a Gaussian in frequency space. High-frequency components are exponentially suppressed, explaining the infinite differentiability of SE-GP sample paths.

For the Matérn-ν kernel: $S(\omega) \propto (2\nu/\ell^2 + \|\omega\|^2)^{-(\nu + d/2)}$, a Student-$t$ in frequency space. The polynomial decay means that high-frequency components, while suppressed, have heavier tails than in the SE case, producing less smooth sample paths.

Random Fourier features (Rahimi and Recht, 2007) approximate a translation-invariant kernel by sampling D frequency vectors $\omega_1, \ldots, \omega_D \sim S(\omega)/\sigma^2$ and constructing features $\phi_j(x) = \sqrt{2/D}[\cos(\omega_j^\top x + b_j)]$ with $b_j \sim \text{Uniform}[0, 2\pi]$:

$$\hat{k}(x, x') = \phi(x)^\top \phi(x') \approx k(x - x').$$

The approximation converges at rate $O(D^{-1/2})$ (by the central limit theorem applied to the average of D i.i.d. cosine features). This reduces the $O(n^3)$ cost of exact GP inference to $O(nD^2 + D^3)$ for $D \ll n$, enabling scalable GP approximations.

0.58.3 7c.3 Multi-Output and Multi-Task Gaussian Processes

In many cognitive modelling applications, multiple related functions must be inferred simultaneously. For example, learning curves across multiple subjects, firing rates of multiple neurons responding to the same stimuli, or predictions of multiple outcome variables. **Multi-output GP** (or **multi-task GP**) extends the single-output framework to vector-valued functions $\mathbf{f} : \mathcal{X} \to \mathbb{R}^T$.

The joint GP prior is specified by a matrix-valued kernel $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{T \times T}$ where $K_{ab}(x, x') = \text{Cov}(f_a(x), f_b(x'))$. A valid such kernel must be positive semidefinite: for all n and all $x_1, \ldots, x_n$, the $nT \times nT$ matrix $[K_{ab}(x_i, x_j)]_{ia,jb}$ must be positive semidefinite.

The **intrinsic coregionalisation model** (ICM) constructs a valid multi-output kernel as:

$$K(x, x') = B \otimes k_0(x, x'),$$

where $B \in \mathbb{R}^{T \times T}$ is a positive semidefinite **coregionalisation matrix** encoding between-output correlations and $k_0 : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a scalar base kernel. Eigendecomposing $B = V \Lambda V^\top$, the T outputs can be expressed as linear combinations of T independent GP "latent functions": $\mathbf{f}(x) = V\mathbf{u}(x)$ where $u_t \sim \mathcal{GP}(0, \lambda_t k_0)$.

For cognitive modelling, multi-task GPs naturally capture individual differences: each participant's learning curve is a draw from a GP that shares a common kernel structure (the population-level covariance) but may differ in mean function or noise level. The joint inference over all participants simultaneously regularises individual estimates toward the population mean, a formal analogue of the hierarchical shrinkage in hierarchical Bayesian models (Section I).

Section *IV*

Section IV — Active Inference and the Free-Energy Principle

0.59 8. The Imperative to Minimise Surprise

The Bayesian Brain hypothesis, as developed in the preceding Section s, is a hypothesis about perception: the brain infers the hidden causes of sensory data using Bayesian inference. But brains do not merely perceive; they act. And they act, in many cases, in order to change the sensory data they receive. This observation — that action and perception are deeply intertwined — is the starting point of Karl Friston's **Free-Energy Principle** and its algorithmic implementation, **Active Inference**.

The central claim of the Free-Energy Principle is that the defining characteristic of any self-organising biological system is its tendency to **minimise variational free energy** — or equivalently, to minimise the surprise of its sensory observations. Every process in such a system, from perception to action to learning, can be understood as an instance of free-energy minimisation.

This is a grand and controversial claim. In this Section , we do not adjudicate its empirical status. We develop the mathematics rigorously, exhibit its internal coherence, and show how it unifies and extends the Bayesian inference framework of the previous Section s.

0.60 9. Variational Free Energy: Formal Development

0.60.1 9.1 Surprise and Its Approximation

In information theory, the **surprise** (or **self-information**) of an observation o under a model with parameters θ is $-\log p(o \mid \theta)$. Minimising expected surprise over time means maximising the model's predictive accuracy: a system that minimises surprise is one that maintains a model of its environment with good predictive power.

For a biological agent with a generative model $p(o, s)$ over observations o and hidden states s, the surprise associated with observing o is $-\log p(o) = -\log \int p(o \mid s)p(s)\, ds$. This marginalisation is, in general, intractable. The variational free energy provides a computable upper bound.

Definition 0.60.1 (Variational Free Energy)**.** Given a generative model $p(o, s)$ and a recognition distribution $q(s)$ over hidden states, the variational free energy is

$$\mathcal{F}(q, o) = \mathbb{E}_{q(s)}[\log q(s) - \log p(o, s)] = \mathrm{KL}(q(s) \| p(s \mid o)) - \log p(o).$$

The second equality follows from the ELBO decomposition we derived in Section II. Since $\mathrm{KL} \geq 0$, we have $\mathcal{F} \geq -\log p(o)$, i.e., $\mathcal{F}$ upper-bounds the surprise. Minimising $\mathcal{F}$ over q simultaneously minimises the KL divergence between q and the true posterior (making the recognition distribution accurate) and improves the predictive accuracy of the generative model.

The free energy can be decomposed in two further illuminating ways:

$$\mathcal{F} = \underbrace{-\mathbb{E}_{q(s)}[\log p(o \mid s)]}_{\text{Expected inaccuracy}} + \underbrace{\mathrm{KL}(q(s) \| p(s))}_{\text{Complexity}}.$$

The first term penalises poor predictions: it is the expected log-likelihood of the observations under the recognition distribution's beliefs about hidden states. The second term penalises complexity: it measures

how much the posterior beliefs $q(s)$ diverge from the prior $p(s)$. Free-energy minimisation thus enacts a precision-weighted trade-off between fitting sensory data and maintaining prior beliefs.

Alternatively:

$$\mathcal{F} = \underbrace{\mathbb{E}_{q(s)}[-\log p(s \mid o)]}_{\text{Posterior energy}} - \underbrace{\mathrm{H}(q(s))}_{\text{Entropy of beliefs}} .$$

Minimising free energy means finding a recognition distribution that places mass near high-probability regions of the posterior (low posterior energy) while remaining as diffuse as uncertainty permits (high entropy).

0.60.2 9.2 Perception, Action, and Learning as Free-Energy Minimisation

In the Active Inference framework, the brain's generative model $p(o, s, a)$ includes actions a as latent variables. An agent can minimise free energy in two ways: (i) **perceptual inference**, by updating the recognition distribution $q(s)$ to better match the posterior given current observations; (ii) **active inference**, by selecting actions a that change future observations to reduce expected free energy.

Formally, actions are selected to minimise **expected free energy** G, which encompasses both expected surprise (epistemic value) and expected divergence from preferred states (pragmatic value):

$$G(\pi) = \mathbb{E}_{q(o,s|\pi)}[\log q(s \mid \pi) - \log p(o, s \mid \pi)] ,$$

where π denotes a policy (a sequence of actions). The expected free energy under a policy decomposes into:

$$G(\pi) = \underbrace{-\mathbb{E}_{q(o|\pi)}[\log p(o)]}_{\text{Expected surprise (pragmatic)}} - \underbrace{\mathrm{I}(o; s \mid \pi)}_{\text{Expected information gain (epistemic)}} .$$

The first term drives the agent toward observations that are likely under its preferred distribution (its goals). The second term drives the agent toward observations that resolve uncertainty about hidden states (curiosity, information seeking). Active Inference thus provides a single mathematical framework for instrumental behaviour (goal-directed action) and exploratory behaviour (information-seeking action) without requiring separate optimality criteria.

Learning in the Active Inference framework corresponds to updating the parameters θ of the generative model to minimise free energy accumulated over time:

$$\Delta\theta \propto -\nabla_\theta \mathcal{F}(\theta) = \mathbb{E}_{q(s)}[\nabla_\theta \log p(o, s \mid \theta)].$$

This is the gradient of a free-energy functional with respect to model parameters — a variational EM-like update that adjusts the generative model to better explain observations. Perception (updating q), action (selecting π), and learning (updating θ) are three routes to the same destination: reduced free energy.

0.60.3 9.3 Predictive Coding: The Neural Implementation

The most influential neural implementation of free-energy minimisation is **predictive coding**, which proposes that the cortex implements a hierarchy of generative models in which each layer sends predictions to the layer below and receives **prediction errors** in return.

In Rao and Ballard's (1999) formulation, a two-level hierarchical generative model with state variables

$r^{(1)}$ and $r^{(2)}$ (at levels 1 and 2) predicts the activity at each level:

$$\text{Level 1: } x = f^{(1)}(r^{(1)}) + \epsilon^{(1)}, \quad \epsilon^{(1)} \sim \mathcal{N}(0, \Sigma^{(1)}),$$

$$\text{Level 2: } r^{(1)} = f^{(2)}(r^{(2)}) + \epsilon^{(2)}, \quad \epsilon^{(2)} \sim \mathcal{N}(0, \Sigma^{(2)}).$$

Free-energy minimisation with respect to $r^{(1)}$ and $r^{(2)}$ yields gradient descent dynamics:

$$\dot{r}^{(1)} = -\frac{\partial \mathcal{F}}{\partial r^{(1)}} \propto (\Sigma^{(1)})^{-1}\epsilon^{(1)} - (J_{f^{(2)}})^\top (\Sigma^{(2)})^{-1}\epsilon^{(2)},$$

where $\epsilon^{(i)}$ are the prediction errors at each level and $J_{f^{(2)}}$ is the Jacobian of $f^{(2)}$. Each state variable is driven by a bottom-up prediction error from below and a top-down signal from above, weighted by precision $(\Sigma^{(i)})^{-1}$. This **precision-weighted prediction error minimisation** is hypothesised to be implemented by the firing rates of pyramidal neurons, with superficial pyramidal cells encoding prediction errors and deep pyramidal cells encoding predictions.

0.61 7e. Gaussian Process Classification: Complete Treatment

0.61.1 7e.1 Laplace Approximation for GP Classification

For binary classification with GP prior $f \sim \mathcal{GP}(0, k)$ and logistic likelihood $p(y_i = 1|f_i) = \sigma(f_i)$, the joint log-posterior is:

$$\log p(\mathbf{f}|X, \mathbf{y}) = \sum_i \log \sigma(y_i f_i) - \frac{1}{2}\mathbf{f}^\top K^{-1}\mathbf{f} - \frac{1}{2}\log\det K - \frac{n}{2}\log(2\pi),$$

where $y_i \in \{-1, +1\}$. This is not Gaussian (the logistic likelihood is non-Gaussian), so the Laplace approximation is used.

Finding the MAP (Newton's method). The gradient and Hessian of the log-posterior:

$$\nabla \log p(\mathbf{f}|X, \mathbf{y}) = \mathbf{t} - K^{-1}\mathbf{f}, \quad \nabla^2 \log p(\mathbf{f}|X, \mathbf{y}) = -W - K^{-1},$$

where $t_i = \frac{\partial}{\partial f_i}\log\sigma(y_i f_i) = y_i\sigma(-y_i f_i)$ and $W = -\text{diag}(\partial^2 \log\sigma(y_i f_i)/\partial f_i^2)$ is a diagonal PSD matrix. Newton's method:

$$\mathbf{f}^{\text{new}} = (K^{-1} + W)^{-1}(W\mathbf{f} + \mathbf{t}).$$

Using the Woodbury identity $(K^{-1} + W)^{-1} = (I - K(I + WK)^{-1}W)K$ and defining $B = I + W^{1/2}KW^{1/2}$ (symmetric positive definite), the MAP can be found in $O(n^3)$ via Cholesky decomposition of B.

Laplace approximation: Approximate $p(\mathbf{f}|X, \mathbf{y}) \approx \mathcal{N}(\hat{\mathbf{f}}, (K^{-1} + W)^{-1})$. The predictive distribution for a new test point x_*:

$$\bar{f}_* = k_*^\top K^{-1}\hat{\mathbf{f}}, \quad \mathbb{V}[f_*] = k(x_*, x_*) - k_*^\top (K + W^{-1})^{-1}k_*,$$

$$p(y_* = 1|x_*, X, \mathbf{y}) = \int \sigma(f_*)\mathcal{N}(f_*; \bar{f}_*, \mathbb{V}[f_*])\, df_* \approx \sigma\left(\frac{\bar{f}_*}{\sqrt{1 + \pi\mathbb{V}[f_*]/8}}\right),$$

where the last integral is approximated by a probit-logistic approximation (exact for a probit likelihood,

approximate for logistic).

0.61.2 7e.2 Expectation Propagation for GP Classification

Expectation Propagation (EP; Minka 2001) provides a more accurate alternative to the Laplace approximation for non-Gaussian likelihoods. EP approximates each likelihood factor $t_i(f_i) = p(y_i|f_i)$ by a Gaussian **site approximation** $\tilde{t}_i(f_i) = \tilde{Z}_i \mathcal{N}(f_i; \tilde{\mu}_i, \tilde{\sigma}_i^2)$.

The EP cavity distribution for site i is the posterior with site i removed:

$$q_{-i}(f_i) \propto q(f_i)/\tilde{t}_i(f_i) \propto \mathcal{N}(f_i; \mu_{-i}, \sigma_{-i}^2),$$

with cavity moments $\sigma_{-i}^{-2} = \tilde{\sigma}_i^{-2} - \Sigma_{ii}^{-1}$ and $\mu_{-i} = \mu - \Sigma_{ii}(\tilde{\sigma}_i^{-2})^{-1}(\mu_i - \tilde{\mu}_i)(\sigma_{-i}^{-2})^{-1}$.

The **EP update** matches the moments of $\hat{p}_i(f_i) \propto t_i(f_i) q_{-i}(f_i)$ to update the site:

$$\tilde{\mu}_i \leftarrow \hat{\mu}_i - \mu_{-i}, \quad \tilde{\sigma}_i^{-2} \leftarrow \hat{\sigma}_i^{-2} - \sigma_{-i}^{-2}, \quad \tilde{Z}_i \leftarrow \hat{Z}_i \mathcal{N}(\hat{\mu}_i; \mu_{-i}, \hat{\sigma}_i^2 + \sigma_{-i}^2)^{-1},$$

where $(\hat{\mu}_i, \hat{\sigma}_i^2)$ are the first two moments of $\hat{p}_i$ (computable analytically for logistic and probit likelihoods via the Mills ratio). EP iterates these updates until convergence, typically in 5–20 passes over the data.

EP is typically more accurate than Laplace (especially in the tails of the predictive distribution) because it matches the first and second moments of each factor rather than approximating the log-posterior quadratically. The EP approximation to the marginal likelihood is also more accurate, making it preferable for kernel hyperparameter optimisation.

0.62 7j. Reproducing Kernel Hilbert Spaces and Kernel Methods

0.62.1 7j.1 The Representer Theorem

The **representer theorem** (Kimeldorf and Wahba 1970; Schölkopf et al., 2001) is one of the most important results in kernel methods. It shows that the solution to a broad class of regularised learning problems lies in the span of the training kernel functions.

Theorem 0.62.1 (Representer Theorem)**.** Let $\mathcal{H}_k$ be the RKHS of kernel k, $\ell : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ any empirical risk functional, and $\Omega : [0, \infty) \to \mathbb{R}$ a strictly monotone increasing regulariser. Then any minimiser of:

$$\hat{f} = \arg\min_{f \in \mathcal{H}_k} \ell(f(x_1), \ldots, f(x_n), y_1, \ldots, y_n) + \Omega(\|f\|_{\mathcal{H}_k}^2)$$

has the form $\hat{f}(x) = \sum_{i=1}^n \alpha_i k(x, x_i)$ for some $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$.

The theorem reduces the infinite-dimensional optimisation in $\mathcal{H}_k$ to a finite-dimensional problem in $\mathbb{R}^n$. For GP regression, the posterior mean $\bar{f}(x) = \mathbf{k}_x^\top K^{-1} \mathbf{y} = \sum_i \alpha_i k(x, x_i)$ is exactly this form with $\alpha_i = (K^{-1}\mathbf{y})_i$.

Kernel ridge regression corresponds to $\ell = \|\mathbf{y} - f(X)\|^2$ and $\Omega = \lambda \|f\|^2$: $\hat{f}(x) = \mathbf{k}_x^\top (K + \lambda I)^{-1} \mathbf{y}$. This is identical to the GP posterior mean with noise variance $\sigma_n^2 = \lambda$, showing that GP regression and kernel ridge regression are the same algorithm in different guises.

0.62.2 7j.2 Support Vector Machines as Maximum Margin Classifiers

The **Support Vector Machine** (SVM; Boser et al., 1992; Vapnik 1995) is a kernel-based classifier that finds the maximum-margin hyperplane in the feature space $\phi(\mathcal{X})$. For binary classification ($y_i \in \{-1, +1\}$), the primal problem:

$$\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C\sum_i \xi_i \quad \text{s.t.} \quad y_i(w^\top \phi(x_i) + b) \geq 1 - \xi_i, \quad \xi_i \geq 0.$$

The dual problem (derived by Lagrangian duality):

$$\max_\alpha \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j k(x_i, x_j) \quad \text{s.t.} \quad 0 \leq \alpha_i \leq C, \quad \sum_i \alpha_i y_i = 0.$$

The prediction for a new point is $\hat{y}(x) = \text{sign}(\sum_i \alpha_i y_i k(x, x_i) + b)$. Only the **support vectors** (training points with $\alpha_i > 0$) contribute to the prediction.

The connection to GP classification: the SVM uses the maximum a posteriori (MAP) estimate under a Laplace likelihood and a GP prior with kernel k, while GP classification (Section 7e) uses the full posterior. SVM is computationally efficient (quadratic programming) but does not provide calibrated probabilities; GP classification provides probabilities but is $O(n^3)$.

0.63 7i. Gaussian Processes in Cognitive Neuroscience

0.63.1 7i.1 GP Decoding of Neural Population Activity

Neural decoding is the inference of stimulus properties or cognitive states from neural activity. A GP decoder places a GP prior over the mapping from neural activity $\mathbf{r} \in \mathbb{R}^N$ (population vector) to stimulus $s \in \mathbb{R}^d$:

$$s = f(\mathbf{r}) + \epsilon, \quad f \sim \mathcal{GP}(0, k_{\text{neural}}), \quad \epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2).$$

The kernel $k_{\text{neural}}(\mathbf{r}, \mathbf{r}')$ encodes the assumption that similar population activity patterns correspond to similar stimuli. The RBF kernel $k_{\text{RBF}}(\mathbf{r}, \mathbf{r}') = \sigma_f^2 \exp(-\|\mathbf{r} - \mathbf{r}'\|^2 / 2\ell^2)$ assumes smooth, non-parametric mapping.

The GP decoder provides:

1. A **point estimate** $\hat{s}(\mathbf{r}^*) = \mathbf{k}_*^\top (K + \sigma_\epsilon^2 I)^{-1} \mathbf{s}_{\text{train}}$ for new population vector $\mathbf{r}^*$.
2. A **confidence interval** $\hat{s}(\mathbf{r}^*) \pm z_{\alpha/2}\sqrt{\mathbb{V}[s(\mathbf{r}^*)]}$.
3. **Uncertainty quantification**: high variance when $\mathbf{r}^*$ is far from training patterns.

Application: decoding of visual orientation. For V1 neurons responding to oriented gratings (orientation $\theta \in [0°, 180°)$), the population vector $\mathbf{r}(\theta)$ is a noisy version of the smooth function $\mathbf{f}(\theta)$ (the tuning curves). The GP decoder with a circular kernel $k(\theta, \theta') = \sigma_f^2 \exp(-2\sin^2(\theta - \theta')/\ell^2)$ correctly accounts for the periodicity of orientation space. The posterior predictive gives optimal Bayesian decoding, achieving the Cramér-Rao bound when the tuning curves are known exactly.

0.63.2 7i.2 GP Smoothing of fMRI BOLD Signals

Functional MRI BOLD signals are noisy, temporally autocorrelated, and spatially smooth. GP smoothing (also known as **Gaussian Process regression for fMRI**) models the BOLD signal at each voxel v as a function of time t:

$$y_v(t) = f_v(t) + \epsilon_v(t), \quad f_v \sim \mathcal{GP}(0, k_{\text{HRF}}), \quad \epsilon_v \sim \mathcal{N}(0, \sigma_v^2),$$

where k_{HRF} is a kernel that encodes the expected shape of the hemodynamic response function (HRF): smooth rise over 4–6 s, peak, smooth return to baseline over 10–12 s. The GP posterior $f_v|\mathbf{y}_v$ provides a denoised estimate of the HRF shape at each voxel, with full uncertainty.

The **marginal likelihood** $p(\mathbf{y}_v|k_{\text{HRF}}, \sigma_v^2)$ provides a model comparison criterion: voxels with HRF shapes consistent with the GP prior (task-activated) have higher marginal likelihood than voxels with flat or noisy responses (inactive). GP-based activation detection is statistically equivalent to the general linear model (GLM) for well-specified HRF, but provides superior performance when the HRF shape is variable across voxels or subjects.

0.63.3 7i.3 Representational Similarity Analysis as a Kernel Method

Representational Similarity Analysis (RSA; Kriegeskorte et al., 2008) compares neural and model representations by comparing their **representational dissimilarity matrices** (RDMs). The neural RDM $D_{ij}^{\text{brain}} = 1 - r(\mathbf{r}_i, \mathbf{r}_j)$ (pairwise neural pattern dissimilarities) is compared to a model RDM D_{ij}^{model} (distances in model representation space) using rank correlation or regression.

The kernel interpretation: $D_{ij}^{\text{brain}} = 1 - k^{\text{brain}}(i,j)/\sqrt{k^{\text{brain}}(i,i)k^{\text{brain}}(j,j)}$ where $k^{\text{brain}}(i,j) = \mathbf{r}_i^\top \mathbf{r}_j/n$ (the linear kernel on neural activity). RSA is thus the comparison of two kernel matrices; it tests whether the two feature spaces (neural and model) have similar geometry.

Kernel RSA (Kornblith et al., 2019) uses the Centered Kernel Alignment (CKA) as the similarity measure:

$$\text{CKA}(K_1, K_2) = \frac{\text{HSIC}(K_1, K_2)}{\sqrt{\text{HSIC}(K_1, K_1)\text{HSIC}(K_2, K_2)}},$$

where $\text{HSIC}(K_1, K_2) = \frac{1}{(n-1)^2}\,\text{tr}(K_1 H K_2 H)$ is the Hilbert-Schmidt Independence Criterion and $H = I - \mathbf{1}\mathbf{1}^\top/n$ is the centering matrix. CKA is invariant to orthogonal transformations and isotropic scaling, making it a principled, invariant measure of representational similarity.

0.64 7h. The Neural Tangent Kernel and Infinite-Width Networks

0.64.1 7h.1 The NTK at Initialization

The **Neural Tangent Kernel** (NTK; Jacot, Gabriel, Hongler 2018) describes the dynamics of gradient descent on neural networks in the infinite-width limit. For a network $f_\theta : \mathcal{X} \to \mathbb{R}$ with parameters θ, the NTK is:

$$K_{\text{NTK}}(x, x') = \left\langle \nabla_\theta f_\theta(x), \nabla_\theta f_\theta(x') \right\rangle = \sum_k \frac{\partial f_\theta(x)}{\partial \theta_k} \frac{\partial f_\theta(x')}{\partial \theta_k}.$$

The NTK measures the similarity between the gradient updates for inputs x and x': if $K_{\text{NTK}}(x, x')$ is large, a gradient step on example x will also improve the prediction on x'.

Theorem 0.64.1 (NTK convergence; Jacot et al. 2018)**.** For an L-layer fully connected network with width $n \to \infty$, the NTK K_{NTK} converges almost surely to a deterministic kernel K^∞ at initialisation, and remains approximately constant throughout training.

The constancy of the NTK during training means that gradient descent on the infinite-width network is equivalent to fitting a kernel regression with kernel K^∞. The network's predictions converge to the kernel regression solution:

$$\hat{f}(x^*) = K^\infty(x^*, X)(K^\infty(X, X) + \lambda I)^{-1}\mathbf{y},$$

where λ is the regularisation (related to the learning rate and step count). This is a GP prediction with kernel K^∞!

0.64.2 7h.2 The NTK as a GP Kernel

The NTK K^∞ satisfies the following recursion for an L-layer ReLU network:

$$K^{(L)}(x, x') = K^{(L-1)}(x, x') \cdot \dot{\Sigma}^{(L)}(x, x') + \Sigma^{(L)}(x, x'),$$

where $\Sigma^{(l)}(x, x') = \mathbb{E}_{f \sim \mathcal{GP}(0, \Sigma^{(l-1)})}[\phi(f(x))\phi(f(x'))]$ is the NNGP kernel at layer l (Section 7b) and $\dot{\Sigma}^{(l)}(x, x') = \mathbb{E}[\phi'(f(x))\phi'(f(x'))]$ is the derivative kernel. The NTK accumulates information from all layers, weighted by the derivative kernels.

This recursion has a fixed point for large L (deep networks): the NTK converges to a specific form that depends on the activation function and network architecture. For ReLU networks, the fixed-point NTK is a **arc-cosine kernel**:

$$K^\infty_{\text{ReLU}}(x, x') = \frac{1}{\pi}\left(\sin\theta + (\pi - \theta)\cos\theta\right)\|x\|\|x'\|,$$

where $\theta = \arccos(x^\top x'/\|x\|\|x'\|)$. This kernel encodes a **Riemannian structure** on the unit sphere that is different from the dot-product or RBF kernel.

0.64.3 7h.3 Finite-Width Corrections and Bayesian Deep Learning

Finite-width networks deviate from the NTK behaviour in two ways:

1. The NTK changes during training (kernel learning).
2. The posterior over network functions is not exactly Gaussian (non-Gaussian corrections).

The **feature learning** regime (finite-width, large learning rate) allows the kernel to change: the network learns representations tailored to the data, unlike the lazy training (small learning rate, infinite width) where the kernel stays fixed. Feature learning is responsible for the generalisation advantage of deep networks over kernel methods in practice.

The **Bayesian deep learning** framework (Neal 1996; Wilson and Izmailov 2020) places a prior over the network parameters and performs Bayesian model averaging over the posterior. For infinite-width networks, this is exactly GP inference with the NNGP or NTK kernel. For finite-width networks, the posterior is non-Gaussian; Markov Chain Monte Carlo (MCMC), variational inference, or the Laplace approximation are used to approximate it. Stochastic Gradient Descent (SGD) with a constant learning rate and weight decay can be interpreted as approximate Bayesian inference (SGLD: Stochastic Gradient Langevin Dynamics), providing a practical approximation to the Bayesian posterior.

0.65 7g. Bayesian Quadrature and Monte Carlo Integration

0.65.1 7g.1 Bayesian Quadrature

Bayesian Quadrature (BQ; Diaconis 1988; O'Hagan 1991) treats numerical integration as a statistical inference problem. Given the integral $\mathcal{Z} = \int f(x)p(x)\,dx$ where f can be evaluated pointwise, BQ places a GP prior over f and infers the distribution over $\mathcal{Z}$ from a finite set of evaluations.

Under the GP prior $f \sim \mathcal{GP}(0, k)$ and evaluations $\mathcal{D} = \{(x_i, f(x_i))\}$, the posterior mean and variance of $\mathcal{Z}$ are:

$$\mathbb{E}[\mathcal{Z}|\mathcal{D}] = \mathbf{z}^\top K^{-1}\mathbf{f}, \qquad \mathbb{V}[\mathcal{Z}|\mathcal{D}] = \iint k(x, x')p(x)p(x')\,dxdx' - \mathbf{z}^\top K^{-1}\mathbf{z},$$

where $z_i = \int k(x_i, x)p(x)\,dx$ is the **kernel mean embedding** and $K_{ij} = k(x_i, x_j)$. The kernel mean embedding z_i is computable analytically for many kernels (e.g., for the RBF kernel with a Gaussian measure: $z_i = \sigma_f^2(2\pi)^{d/2}\det(\ell^2 I + \Sigma)^{-1/2}\exp(-\frac{1}{2}(x_i - \mu)^\top(\ell^2 I + \Sigma)^{-1}(x_i - \mu))$).

BQ reduces to standard Monte Carlo when the GP prior is flat (infinite-length-scale kernel). For smooth functions, BQ achieves **faster convergence rates** than Monte Carlo: $O(n^{-2s/(2s+d)})$ for f in a Sobolev space of order s, compared to the Monte Carlo rate $O(n^{-1/2})$ independent of smoothness. This faster rate becomes decisive for high-dimensional integrals over smooth functions, making BQ the method of choice when f is expensive to evaluate and smooth.

0.65.2 7g.2 Maximum Mean Discrepancy and Kernel Methods for Distributions

The **Maximum Mean Discrepancy** (MMD; Gretton et al., 2012) is a kernel-based divergence between two distributions P and Q:

$$\text{MMD}^2(P, Q; k) = \mathbb{E}_{x,x'\sim P}[k(x, x')] - 2\mathbb{E}_{x\sim P, y\sim Q}[k(x, y)] + \mathbb{E}_{y,y'\sim Q}[k(y, y')].$$

MMD$= 0$ iff $P = Q$ (for a characteristic kernel like the RBF kernel). The empirical estimator $\widehat{\text{MMD}}^2$ from n samples from P and m from Q has expectation $\text{MMD}^2(P, Q)$ and variance $O(1/n + 1/m)$.

Kernel two-sample test. Under $H_0 : P = Q$, the MMD statistic follows (asymptotically) the distribution of a weighted sum of chi-squared variables; the test rejects when $\widehat{\text{MMD}}^2 > c_\alpha$ (critical value at level α). The MMD test is consistent (power $\to 1$ as $n \to \infty$ for any $P \neq Q$) and can detect high-dimensional differences that marginal or pairwise tests miss.

Cognitive application: comparing neural representations. To test whether two groups of participants (e.g., experts vs novices) have different neural representations of the same stimuli, compute the MMD between the distributions of fMRI activation vectors:

$$\widehat{\text{MMD}}^2 = \frac{1}{n^2}\sum_{i,j} k(\mathbf{x}_i, \mathbf{x}_j) - \frac{2}{nm}\sum_{i,j} k(\mathbf{x}_i, \mathbf{y}_j) + \frac{1}{m^2}\sum_{i,j} k(\mathbf{y}_i, \mathbf{y}_j),$$

where $\mathbf{x}_i$ are expert activations and $\mathbf{y}_j$ are novice activations. A significant MMD indicates that the two groups process the stimuli differently in a kernel-measurable sense, even if the mean activations are similar.

0.66 7f. Deep Kernel Learning and Neural Architecture Priors

0.66.1 7f.1 Deep Kernel Learning

Deep Kernel Learning (DKL; Wilson et al., 2016) combines the representation learning of deep neural networks with the principled uncertainty quantification of Gaussian processes. The idea: replace the fixed feature space in the GP kernel with a learned deep feature mapping.

For a DNN $\phi_w : \mathcal{X} \to \mathbb{R}^D$ with parameters w, the DKL kernel is:

$$k_{\text{DKL}}(x, x') = k_0(\phi_w(x), \phi_w(x')),$$

where k_0 is a base kernel (e.g., RBF or spectral mixture) in the feature space. Training jointly optimises w and the GP hyperparameters by maximising the marginal likelihood:

$$\log p(\mathbf{y}|X, w, \psi) = -\frac{1}{2}\mathbf{y}^\top K_{\text{DKL}}^{-1}\mathbf{y} - \frac{1}{2}\log\det K_{\text{DKL}} - \frac{n}{2}\log 2\pi,$$

where K_{DKL} is the kernel matrix computed from the DNN features.

DKL inherits the benefits of both worlds: the DNN learns a powerful, task-relevant feature representation (handling high-dimensional inputs like images), while the GP provides exact posterior uncertainty over function values given the learned features. The marginal likelihood objective prevents overfitting to the training data: the complexity penalty $-\frac{1}{2}\log\det K$ discourages unnecessarily complex (large) kernels.

The computational cost is $O(n^3)$ for exact inference, but sparse approximations (inducing points, random features) reduce this. For classification, DKL with a GP classifier provides well-calibrated predictive probabilities even on out-of-distribution inputs, a known failure mode of standard DNNs.

0.66.2 7f.2 GP Prior over Neural Architecture Space

Neural Architecture Search (NAS) with GP priors treats the architecture space $\mathcal{A}$ (the space of all possible network configurations) as the domain of a Gaussian process. The objective is to find the architecture $a^* \in \mathcal{A}$ that maximises validation performance $f(a)$, where f is expensive to evaluate (each evaluation requires training a network).

Bayesian optimisation for NAS. Place a GP prior over f: $f \sim \mathcal{GP}(0, k_{\mathcal{A}})$ where $k_{\mathcal{A}}$ is a kernel on architecture space. After evaluating T architectures $\{a_t, f(a_t)\}_{t=1}^T$, the posterior $f|\{a_t, f(a_t)\}$ is a GP. The next architecture to evaluate is chosen by maximising an **acquisition function** $\alpha(a)$ that balances exploration and exploitation:

Expected Improvement (EI):

$$\alpha_{\text{EI}}(a) = \mathbb{E}[\max(f(a) - f^+, 0)],$$

where $f^+ = \max_t f(a_t)$ is the best observed performance. For a GP posterior with mean $\mu(a)$ and variance $\sigma^2(a)$:

$$\alpha_{\text{EI}}(a) = (\mu(a) - f^+)\Phi\left(\frac{\mu(a) - f^+}{\sigma(a)}\right) + \sigma(a)\phi\left(\frac{\mu(a) - f^+}{\sigma(a)}\right),$$

where Φ and ϕ are the Gaussian CDF and PDF. EI is zero for $\sigma = 0$ (certainty about performance) and maximised at the point where the posterior places high probability on exceeding f^+.

The kernel $k_{\mathcal{A}}$ on architecture space must handle the discrete, structured nature of architectures (number of layers, types of operations, connection patterns). **Path kernels**, **graph kernels**, and **Weisfeiler-Leman kernels** measure structural similarity between architectures; the GP posterior then interpolates performance across structurally similar architectures.

0.67 7d. Gaussian Processes for Temporal Cognition

0.67.1 7d.1 GP Models of Learning Curves

Learning curves describe how performance changes over practice. A fundamental empirical regularity is the **power law of practice**: performance (measured by reaction time RT or error rate) decreases as a power law of the number of practice trials n:

$$RT(n) = a + bn^{-c}, \qquad c \in (0, 1].$$

This power law is consistent with a hierarchical Bayesian model in which the learner acquires increasingly general knowledge (chunks, rules, prototypes) over trials: each new chunk reduces the time for future trials by a fixed fraction, producing exponential decay in chunk-level RT and power-law decay in observed RT (sum of decaying exponentials $\approx$ power law).

GP models of learning curves (Ritter and Schooler 2001; Newell and Rosenbloom 1981) place a GP prior directly on the function $f(n) = \mathbb{E}[RT|\text{trial} = n]$:

$$f \sim \mathcal{GP}(m, k_{\text{per-epoch}}), \qquad RT_i = f(n_i) + \epsilon_i,$$

where $m(n) = a + bn^{-c}$ is a parametric mean function encoding the power law prior, and k is a residual kernel capturing deviations from the power law. The GP posterior $f|\mathbf{RT}$ provides a calibrated uncertainty estimate over the learning curve, propagating both parameter uncertainty (in a, b, c) and residual uncertainty (in k).

The GP framework naturally handles **missing data**: if some trials are unobserved (e.g., the participant took a break), the GP posterior interpolates smoothly over the gap, with uncertainty that increases with the size of the gap. This is in contrast to fixed-function models, which must either ignore the gap or impute specific values.

0.67.2 7d.2 GP-SSM: Gaussian Process State-Space Models

For cognitive time series (EEG, fMRI, reaction times), the data are sequential and exhibit temporal autocorrelation. The **GP state-space model** (GP-SSM; Turner et al., 2010; Frigola et al., 2014) places a GP prior on the **transition function** of a state-space model:

$$s_{t+1} = f(s_t) + \epsilon_t, \quad x_t = g(s_t) + \eta_t,$$

where $f \sim \mathcal{GP}(0, k_f)$ is the transition function and $g \sim \mathcal{GP}(0, k_g)$ is the emission function. Both f and g are unknown and inferred from data simultaneously with the latent states $s_{1:T}$.

This is a highly flexible non-parametric model: it does not assume a specific form for the dynamics (e.g., linear, oscillatory) but infers it from data. The posterior over f and g captures uncertainty about the system's dynamical structure. Inference is performed using variational inference or particle smoothers.

For cognitive modelling, GP-SSMs provide a data-driven framework for modelling neural dynamics without committing to a specific dynamical architecture. Given EEG recordings from a cognitive task, the GP-SSM infers the latent neural state trajectory and the dynamics that govern it, enabling hypothesis-testing about cognitive state transitions (e.g., attention shifts, decision-making).

0.67.3 7d.3 Kernel Hyperparameter Learning and Model Selection

GP hyperparameters $\boldsymbol{\psi}$ (length-scale ℓ, signal variance σ^2, noise variance σ_n^2) are typically estimated by **marginal likelihood maximisation**:

$$\boldsymbol{\psi}^* = \arg\max_{\boldsymbol{\psi}} \log p(\mathbf{y}|X, \boldsymbol{\psi}),$$

$$\log p(\mathbf{y}|X, \boldsymbol{\psi}) = -\frac{1}{2}\mathbf{y}^\top (K_{XX} + \sigma_n^2 I)^{-1}\mathbf{y} - \frac{1}{2}\log\det(K_{XX} + \sigma_n^2 I) - \frac{n}{2}\log 2\pi.$$

The three terms have interpretations: the **data fit** $-\frac{1}{2}\mathbf{y}^\top K^{-1}\mathbf{y}$ rewards the model for predicting the observed outputs; the **complexity penalty** $-\frac{1}{2}\log\det K$ penalises complex models (large kernels); the constant ensures normalisation.

The marginal likelihood gradient with respect to a hyperparameter ψ_j:

$$\frac{\partial \log p}{\partial \psi_j} = \frac{1}{2}\operatorname{tr}\left((K^{-1}\mathbf{y}\mathbf{y}^\top K^{-1} - K^{-1})\frac{\partial K}{\partial \psi_j}\right).$$

This can be computed efficiently using Cholesky decomposition of K in $O(n^2)$ per gradient step (after the $O(n^3)$ Cholesky). Gradient ascent on $\log p$ over $\boldsymbol{\psi}$ is the standard method for hyperparameter learning.

Bayesian hyperparameter optimisation. Full Bayesian treatment places priors on hyperparameters $p(\boldsymbol{\psi})$ and integrates them out:

$$p(\mathbf{y}|X) = \int p(\mathbf{y}|X, \boldsymbol{\psi})\, p(\boldsymbol{\psi})\, d\boldsymbol{\psi}.$$

This is intractable in general, but Monte Carlo over $\boldsymbol{\psi}$ is effective when the hyperparameter space is low-dimensional (≤ 10). Bayesian hyperparameter optimisation provides better calibration (correct uncertainty quantification even in small-data regimes) at higher computational cost.

0.68 9j. Active Learning and Experimental Design

0.68.1 9j.1 Optimal Experimental Design as Active Inference

Optimal experimental design (OED) chooses experiments to maximise the information gained about the parameters of a model. In the Bayesian framework, the optimal experiment x^* maximises the expected information gain (mutual information between the experiment's outcome and the parameters):

$$x^* = \arg\max_x \mathrm{I}(\theta; y|x) = \arg\max_x \mathbb{E}_{p(y|x)}[\mathrm{KL}(p(\theta|y, x)\|p(\theta))].$$

This is the **epistemic value** of Active Inference: the experiment that maximally reduces uncertainty about the parameters.

For the special case of a linear model $y = X\theta + \epsilon$ with $\theta \sim \mathcal{N}(0, \Lambda_0^{-1})$ and $\epsilon \sim \mathcal{N}(0, \sigma^2)$, the expected information gain after observing y at design x is:

$$\mathrm{I}(\theta; y|x) = \frac{1}{2}\log(1 + \sigma^{-2} x^\top \Lambda_0^{-1} x).$$

The optimal design maximises $x^\top \Lambda_0^{-1} x$: points in the directions of highest prior uncertainty. This is **D-optimal design**: maximise the determinant of the Fisher information matrix $\mathcal{I}(\theta) = \sigma^{-2}\sum_{i=1}^n x_i x_i^\top$, which maximises the Section of the confidence ellipsoid for θ.

Adaptive experimental design. In cognitive science, experiments can be adapted in real-time: choose each trial based on current posterior beliefs about the participant's parameters. The **Adaptive Bayesian Approach** (Kontsevich and Tyler 1999) for psychophysics: estimate the psychometric function $\psi(\theta; c)$ (probability of "yes" at contrast c, parametrised by threshold and slope θ) and choose the contrast c^* that maximises the expected information gain about θ. This requires $\sim 50\%$ fewer trials than traditional staircase methods for the same accuracy.

0.68.2 9j.2 BOED with Neural Networks

Bayesian Optimal Experimental Design with neural networks (BOED; Foster et al., 2019) amortises the OED computation using trained neural networks. For each design x, the expected information gain (EIG) involves an intractable double integral:

$$\mathrm{EIG}(x) = \int\int p(y|x,\theta)p(\theta)\log\frac{p(\theta|y,x)}{p(\theta)}\,d\theta\,dy.$$

BOED trains a neural network to estimate the mutual information directly from samples, using the **MINE** (Mutual Information Neural Estimation; Belghazi et al., 2018) or **VNCE** (Variational Noise Contrastive Estimation) estimators. After training, the EIG can be evaluated for any design x in milliseconds, enabling real-time adaptive experimentation.

0.69 9i. Variational Free Energy: Biological Interpretations

0.69.1 9i.1 Free Energy and Surprise

The variational free energy $\mathcal{F}(q, \mathbf{s}) = \mathbb{E}_q[\ln q(\tilde{\mu})] - \mathbb{E}_q[\ln p(\tilde{\mu}, \tilde{s})]$ bounds the surprisal (negative log-evidence) $-\ln p(\tilde{s})$:

$$\mathcal{F}(q, \tilde{s}) = -\ln p(\tilde{s}) + \mathrm{KL}(q(\tilde{\mu}) \| p(\tilde{\mu}|\tilde{s})) \geq -\ln p(\tilde{s}).$$

For biological systems, surprisal $-\ln p(\tilde{s})$ has a direct physical interpretation: it is the **self-information** of the observed sensory state under the agent's model. Low surprisal means the sensory state is expected (consistent with the model); high surprisal means the sensory state is unexpected. A biological system at non-equilibrium steady state has low mean surprisal: it occupies states consistent with being alive.

Free energy decomposition. The variational free energy decomposes into three terms:

$$\mathcal{F} = \underbrace{\mathbb{E}_q[\text{Energy}]}_{\text{internal energy}} - \underbrace{H(q)}_{\text{entropy}} = \underbrace{-\ln p(\tilde{s})}_{\text{surprisal}} + \underbrace{\text{KL}(q \| p(\tilde{\mu}|\tilde{s}))}_{\text{inaccuracy}}.$$

The first decomposition (energy minus entropy) is the thermodynamic free energy; the second (surprisal plus inaccuracy) is the information-theoretic decomposition. Minimising free energy means simultaneously minimising surprisal (staying in expected states) and minimising the KL divergence (having accurate beliefs about causes).

0.69.2 9i.2 Precision and Synaptic Gain

Precision is the inverse variance (reliability) of a prediction error: $\Pi = \Sigma^{-1}$. In the predictive coding equations, each level's dynamics are:

$$\tau \dot{\mu}^{(l)} = -\Pi^{(l)} \xi^{(l)} + J_{g^{(l)}}^{\top} \Pi^{(l+1)} \xi^{(l+1)},$$

where $\xi^{(l)} = \mu^{(l)} - g^{(l)}(\mu^{(l+1)})$ is the prediction error and $\Pi^{(l)}$ weights its influence. Higher precision $\Pi^{(l)}$ means the prediction error at level l has more influence on updating beliefs at level l.

Biologically, precision is modulated by **neuromodulators**: dopamine (reward salience), acetylcholine (sensory precision), noradrenaline (global arousal). The FEP interpretation (Friston 2009): neuromodulators set the gain on prediction error units, effectively setting the precision of sensory vs. prior signals. This is the mathematical account of **attentional modulation**: attention increases the precision of attended sensory channels, causing attended stimuli to have more influence on beliefs.

Clinical prediction. Dysregulated precision underlies several psychiatric disorders (Fletcher and Frith 2009):

- **Schizophrenia**: aberrantly high precision on weak sensory signals generates perceptual aberrations (hallucinations: treating noise as a high-precision signal).
- **Autism Spectrum Disorder**: overweighting of sensory precision relative to prior precision generates sensory hypersensitivity and resistance to top-down context (insistence on sameness).
- **Depression**: underweighting of sensory precision relative to prior precision generates over-reliance on prior beliefs (negative cognitive schemas dominate perception).

0.69.3 9i.3 Interoception and Homeostasis as Active Inference

Interoception — the sensing of the body's internal state — provides a continuous stream of information about the body's metabolic, autonomic, and visceral state. In the Active Inference framework, interoceptive signals are the sensory observations for a body-level generative model, and autonomic regulation is the action (muscle tone, heart rate, respiration) that minimises interoceptive prediction errors.

The interoceptive generative model predicts body states from metabolic context: after exercise, heart rate should be elevated; after a meal, blood glucose should rise. Interoceptive prediction errors (actual minus predicted heart rate, etc.) drive autonomic reflexes via the vagus nerve and hypothalamic-pituitary-adrenal (HPA) axis. Chronic interoceptive prediction errors signal allostatic failure: the body cannot maintain the predicted internal state.

Mood as interoceptive precision. Mood states have been formalised as the expected precision of interoceptive prediction errors (Seth and Friston 2016): positive affect corresponds to confident prediction of positive body states (high precision on expected-positive interoceptive signals); negative affect corresponds to the opposite. **Interoceptive Active Inference** — the active regulation of body states to match interoceptive priors — is the neural basis of emotion regulation, providing a mathematical account of how cognitive states modulate bodily responses.

0.70 9h. Continuous-Time Active Inference

0.70.1 9h.1 Generalised Coordinates and Continuous-Time Inference

Discrete-time predictive coding and Active Inference assume that beliefs are updated at discrete time steps. In **continuous-time active inference** (Friston et al., 2010), beliefs are updated continuously as sensory streams arrive, using **generalised coordinates of motion**.

Generalised coordinates represent the state $\tilde{\mu}$ as an infinite-dimensional vector $[\mu, \mu', \mu'', \ldots]$ where $\mu^{(k)}$ is the k-th time derivative of the state. Truncating at order p: $\tilde{\mu} = [\mu, \mu', \ldots, \mu^{(p)}]$. The shift operator $\mathbf{D}$ maps $[\mu, \mu', \ldots] \mapsto [\mu', \mu'', \ldots]$; in generalised coordinates, the equation of motion becomes:

$$\dot{\tilde{\mu}} = \mathbf{D}\tilde{\mu} - \kappa \nabla_{\tilde{\mu}} \mathcal{F}(\tilde{\mu}, \tilde{s}),$$

where $\mathbf{D}\tilde{\mu}$ is the **prediction** (the state should follow its own motion) and the second term is the **correction** from the free energy gradient (κ is a gain parameter).

The free energy in generalised coordinates is:

$$\mathcal{F}(\tilde{\mu}, \tilde{s}) = \frac{1}{2} \tilde{\boldsymbol{\epsilon}}_s^\top \tilde{\Pi}_s \tilde{\boldsymbol{\epsilon}}_s + \frac{1}{2} \tilde{\boldsymbol{\epsilon}}_v^\top \tilde{\Pi}_v \tilde{\boldsymbol{\epsilon}}_v,$$

where $\tilde{\boldsymbol{\epsilon}}_s = \tilde{s} - \tilde{g}(\tilde{\mu})$ is the sensory prediction error in generalised coordinates and $\tilde{\boldsymbol{\epsilon}}_v = \mathbf{D}\tilde{\mu} - \tilde{f}(\tilde{\mu})$ is the dynamic prediction error (how much does the state's motion deviate from the predicted motion?). The precision matrices $\tilde{\Pi}_s$ and $\tilde{\Pi}_v$ are block-diagonal in generalised coordinates.

0.70.2 9h.2 Filtering as Free Energy Minimisation

The generalised Bayesian filter (GBF) for continuous-time state-space models is derived from free energy minimisation in generalised coordinates. For a model:

$$\dot{x} = f(x, v) + \omega_x, \quad y = g(x, v) + \omega_y,$$

with Gaussian noise $\omega_x \sim \mathcal{N}(0, S_x)$ and $\omega_y \sim \mathcal{N}(0, S_y)$, the GBF computes the posterior over the state trajectory $x(t)$ and external input $v(t)$ given the observation trajectory $y(t)$.

The Kalman-Bucy filter (the continuous-time Kalman filter) is the special case for linear f and g:

$$\dot{\hat{x}} = A\hat{x} + K(y - H\hat{x}), \quad K = PH^\top S_y^{-1},$$

$$\dot{P} = AP + PA^\top + S_x - PH^\top S_y^{-1} HP,$$

where K is the Kalman-Bucy gain and P is the error covariance. The GBF reduces to the Kalman-Bucy filter for linear models and extends to nonlinear models via generalised coordinates.

0.70.3 9h.3 Action as Proprioceptive Prediction Error Minimisation

In Active Inference, actions minimise the free energy by changing the world to match sensory predictions, rather than by changing beliefs to match sensory observations. For continuous-time motor control:

$$\dot{a} = -\kappa_a \frac{\partial \mathcal{F}}{\partial \tilde{s}} \cdot \frac{\partial \tilde{s}}{\partial a},$$

where a is the motor command (e.g., joint torques) and $\partial \tilde{s}/\partial a$ is the Jacobian of sensory states with respect to action. Actions flow in the gradient direction that decreases the sensory prediction error: if the proprioceptive prediction is "arm at position x^*" and the current position is x, actions drive the arm toward x^*.

This is formally equivalent to **model predictive control** (MPC): the agent computes the action that minimises the predicted future prediction error over a finite horizon. The difference from classical MPC: in Active Inference, the setpoint x^* is determined by the prior beliefs (goal states encoded in the generative model), not by an external cost function. Goals are encoded as **prior preferences** in the generative model: $P(\tilde{s}) \propto \exp(-E(\tilde{s}))$ where E is a potential function whose minima correspond to goal states.

0.71 9g. Perception as Inference: Bayesian Brain Hypothesis

0.71.1 9g.1 The Bayesian Brain Hypothesis

The **Bayesian Brain Hypothesis** (Knill and Pouget 2004; Körding and Wolpert 2004) is the general claim that the brain represents and manipulates probabilistic beliefs about the world, performing Bayesian inference to combine prior knowledge with sensory evidence. This hypothesis unifies a disparate range of perceptual and motor phenomena under a common computational framework.

Cue combination. When multiple sensory cues provide information about the same environmental variable (e.g., visual and proprioceptive information about hand position), the optimal combination is a precision-weighted average:

$$\hat{x} = \frac{\sigma_V^{-2} x_V + \sigma_P^{-2} x_P}{\sigma_V^{-2} + \sigma_P^{-2}}, \qquad \text{Var}(\hat{x}) = \frac{1}{\sigma_V^{-2} + \sigma_P^{-2}},$$

where x_V and x_P are the visual and proprioceptive estimates, and σ_V^2, σ_P^2 are their respective variances. The combined estimate weights more reliable cues (lower variance) more heavily. Körding and Wolpert (2004) confirmed this prediction: human subjects combined visual and proprioceptive information in a statistically optimal way that matched the Bayesian prediction.

Prior effects on perception. Weiss et al. (2002) showed that the **slow motion prior** (the brain expects objects to move slowly) explains the aperture problem in motion perception: a grating moving behind an aperture appears to move perpendicularly to its orientation, because this is the slowest motion consistent with the visual input. Formally, $P(v) \propto \exp(-|v|^2/2\sigma_{\text{prior}}^2)$ is a Gaussian prior centred at zero velocity; the posterior favours the slow-but-consistent motion.

0.71.2 9g.2 Predictive Coding: Derivation from First Principles

We derive the predictive coding equations from the variational free energy minimisation principle (Section 9.1). Consider a two-level generative model:

$$P(v) = \mathcal{N}(v; 0, \Sigma_v), \quad P(u|v) = \mathcal{N}(u; g(v), \Sigma_u), \quad P(y|u) = \mathcal{N}(y; h(u), \Sigma_y),$$

where v are high-level causes, u are intermediate representations, y are observations, g and h are nonlinear generative functions, and Σ_v, Σ_u, Σ_y are precision matrices ($\Pi_v = \Sigma_v^{-1}$, etc.).

The mean-field free energy with Gaussian $q(v) = \mathcal{N}(\mu_v, \Sigma_{qv})$ and $q(u) = \mathcal{N}(\mu_u, \Sigma_{qu})$:

$$\mathcal{F} = \frac{1}{2}\left[\Pi_v \|\mu_v\|^2 + \Pi_u \|\mu_u - g(\mu_v)\|^2 + \Pi_y \|y - h(\mu_u)\|^2\right] + \text{const.}$$

(Ignoring the covariance terms for simplicity, treating it as a point-mass approximation — the MAP estimate.)

Gradient descent on $\mathcal{F}$:

$$\tau \dot{\mu}_v = -\Pi_v \mu_v + J_g(\mu_v)^\top \Pi_u (\mu_u - g(\mu_v)),$$
$$\tau \dot{\mu}_u = -\Pi_u (\mu_u - g(\mu_v)) + J_h(\mu_u)^\top \Pi_y (y - h(\mu_u)).$$

Define **prediction errors**:

$$\xi_v = \Pi_v^{1/2} \mu_v, \quad \xi_u = \Pi_u^{1/2}(\mu_u - g(\mu_v)), \quad \xi_y = \Pi_y^{1/2}(y - h(\mu_u)).$$

The dynamics in terms of prediction errors:

$$\tau \dot{\mu}_v = -\xi_v + J_g^\top \Pi_u^{1/2} \xi_u, \qquad \tau \dot{\mu}_u = -\xi_u + J_h^\top \Pi_y^{1/2} \xi_y.$$

Each level's state is driven by: (1) the downward prediction error ξ_v or ξ_u (how much the current estimate deviates from the higher-level prediction) and (2) the upward prediction error (the derivative of the error at the level below, passed up through the Jacobian).

This is the **predictive coding architecture**: higher-level units generate predictions that are subtracted from lower-level inputs; the residual prediction errors propagate upward and drive updates to higher-level beliefs. The architecture is local and parallel: each unit needs only information from immediately adjacent levels.

0.71.3 9g.3 Explaining Away and Explaining Too Much

A key prediction of the Bayesian brain hypothesis is **explaining away**: when one cause fully explains an observation, the posterior probability of alternative causes decreases. For example, if motion is explained by a moving observer (self-motion), the posterior probability of external object motion decreases.

Formally, for competing causes c_1 (self-motion) and c_2 (external motion) with shared effect y (retinal flow):

$$P(c_1, c_2|y) \propto P(y|c_1, c_2) P(c_1) P(c_2).$$

If $P(y|c_1, c_2) = P(y|c_1 + c_2)$ (the effect is the sum of causes), then for observed $y = y_0$: the constraint

$c_1 + c_2 \approx y_0$ creates a negative correlation between c_1 and c_2 in the posterior. Knowing $c_1 = y_0$ (full self-motion explanation) drives $P(c_2 \approx 0)$ to 1: c_2 is explained away.

Neurally, explaining away is implemented by top-down inhibition: a higher-level unit encoding cause c_1 sends predictions downward that suppress the responses of lower-level units sensitive to the shared effect y. If c_1 is well-established, the top-down suppression nearly cancels the bottom-up input, leaving a small residual error — the neural signature of a well-predicted input (low prediction error, low neural activity).

Explaining too much (hyperpriors and hallucinations). If the precision of priors is too high relative to sensory evidence, the brain may see patterns that are not there: **hallucinations**. Mathematically, if $\Pi_v \gg \Pi_y$ (prior dominates likelihood), the posterior μ_v stays near the prior mean regardless of the sensory evidence y. In predictive coding, this corresponds to top-down predictions overwhelmingly suppressing bottom-up inputs, with very small residual errors. The brain "sees" what it expects rather than what is there.

Psychotic disorders (schizophrenia, bipolar disorder) have been proposed to involve dysregulated precision: aberrantly high prior precision (or low sensory precision) gives rise to the positive symptoms (hallucinations, delusions) of psychosis. Antipsychotic medications (D2 receptor antagonists) may work by reducing dopaminergic precision signals, restoring the balance between priors and sensory evidence.

0.72 9f. Discrete Active Inference: A Worked Example

0.72.1 9f.1 The T-Maze Task

The **T-maze** is the canonical benchmark for Active Inference. An agent faces a T-shaped maze with four locations: a central start (location 1), a left arm (location 2), a right arm (location 3), and a cue location (location 4). One of the arms contains reward; the other is empty. The cue location provides a signal indicating which arm contains reward.

Generative model. Hidden states $s_t = (s_t^{\text{loc}}, s_t^{\text{ctx}})$ where $s^{\text{loc}} \in \{1, 2, 3, 4\}$ is the current location and $s^{\text{ctx}} \in \{\text{left-rewarded}, \text{right-rewarded}\}$ is the context. The context is stable ($P(s_{t+1}^{\text{ctx}}|s_t^{\text{ctx}}) = I$). The location transitions depend on action $a_t \in \{\text{stay, go-left, go-right, go-cue}\}$.

Observations $o_t = (o_t^{\text{loc}}, o_t^{\text{rew}})$: location observation (unambiguous) and reward observation (reward / no-reward / cue-signal / neutral). The likelihood matrix A (observations given states) encodes:

- From location 2 (left arm): reward with probability P_R if context = left-rewarded, otherwise no-reward.
- From location 4 (cue): cue-signal-left if context = left-rewarded, cue-signal-right otherwise.

0.72.2 9f.2 Policy Evaluation via Expected Free Energy

At time $t = 0$ (start), the agent evaluates all possible two-step policies $\pi = (a_0, a_1)$. For each policy, the expected free energy is:

$$\mathbf{G}(\pi) = \sum_{\tau=0}^{1} \mathbb{E}_{Q(o_\tau, s_\tau|\pi)}[\log Q(s_\tau|\pi) - \log P(s_\tau|\pi) - \log P(o_\tau)] \,.$$

Epistemic value. The policy $\pi = (\text{go-cue}, \text{go-left/right based on cue})$ has high epistemic value: visiting the cue location resolves the uncertainty about the context, enabling confident choice of the reward arm in the second step. The mutual information $\mathbf{I}(o_0^{\text{rew}}; s_0^{\text{ctx}})$ is maximised by visiting the cue.

Pragmatic value. The policy $\pi = (\text{go-left}, \text{stay})$ has high pragmatic value only if the agent already believes with high probability that left is rewarded. For a flat prior $P(s^{\text{ctx}}) = (0.5, 0.5)$, pragmatic value alone recommends random arm selection; combined with epistemic value, the optimal policy is to visit the cue first.

Policy selection. The probability of selecting policy π is:

$$P(\pi) = \sigma(-\gamma \mathbf{G}(\pi)) \propto \exp(-\gamma \mathbf{G}(\pi)),$$

where γ is the inverse temperature (precision of action selection). For $\gamma \to \infty$, the agent selects the policy with minimum $\mathbf{G}$ (greedy exploitation); for $\gamma \to 0$, it selects uniformly (pure exploration).

Numerical example. With $P_R = 0.9$, $\gamma = 4$, flat prior, and log preferences $\log P(o) = 2$ for reward and 0 otherwise:

$$\mathbf{G}(\text{cue-first}) \approx -\,\mathbf{I}(\text{cue}; \text{context}) - P(\text{reward}|\text{cue-first}) \cdot 2 \approx -1.0 - 0.9 \cdot 2 = -2.8,$$

$$\mathbf{G}(\text{left-direct}) \approx 0 - 0.5 \cdot 2 = -1.0.$$

Since $\mathbf{G}(\text{cue-first}) < \mathbf{G}(\text{left-direct})$, the agent selects cue-first with probability $\sigma(4 \cdot 1.8) \approx 0.999$.

0.72.3 9f.3 Belief Updating During Execution

As the agent executes its policy, beliefs are updated at each time step. After visiting the cue and observing cue-signal-left, the posterior over context is:

$$Q(s^{\text{ctx}} = \text{left-rewarded}|o^{\text{rew}} = \text{cue-left}) = \frac{P(\text{cue-left}|\text{left-rewarded})P(\text{left-rewarded})}{P(\text{cue-left})} = \frac{1 \cdot 0.5}{0.5} = 1.$$

The context is now known with certainty. The second-step action selects left arm, and the agent receives reward with probability $P_R = 0.9$.

The T-maze demonstrates the core Active Inference claim: the agent's behaviour is not reward-maximisation (which would choose the arm at random initially, then update based on feedback) but **epistemic foraging**: actively seeking information that resolves uncertainty before committing to a reward-seeking action. This matches the empirical observation that rodents in T-maze tasks **reliably visit the cue arm first** before approaching the reward arm, an exploratory behaviour that cannot be explained by reward-maximisation alone.

0.73 9b. POMDPs and Active Inference: The Formal Connection

0.73.1 9b.1 Partially Observable Markov Decision Processes

A **Partially Observable Markov Decision Process** (POMDP) is a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{O}, T, E, R, \gamma)$ where $\mathcal{S}$ is the state space, $\mathcal{A}$ the action space, $\mathcal{O}$ the observation space, $T(s'|s, a)$ the transition model, $E(o|s, a)$ the emission model, $R(s, a)$ the reward function, and $\gamma \in (0, 1)$ the discount factor. The agent cannot

observe the state directly; it maintains a **belief state** $b_t = p(s_t|o_{1:t}, a_{1:t-1})$, a sufficient statistic for optimal decision-making.

The Bellman equation for the optimal value function $V^*(b)$ of a POMDP is:

$$V^*(b) = \max_{a \in \mathcal{A}} \left[R(b,a) + \gamma \sum_{o \in \mathcal{O}} p(o|b,a)\, V^*(\tau(b,a,o)) \right],$$

where $R(b,a) = \sum_s b(s) R(s,a)$ is the expected reward and $\tau(b,a,o)$ is the updated belief after taking action a and observing o (computed by Bayes' rule). The optimal value function is a piecewise-linear convex function of the belief state (Sondik 1971); the PBVI and HSVI algorithms compute approximate solutions by maintaining finite sets of α-vectors.

0.73.2 9b.2 Active Inference as Deep POMDP

Active Inference reformulates the POMDP objective as free-energy minimisation. In the discrete-state, discrete-time Active Inference framework (da Costa et al., 2020), the generative model specifies:

$$P(o_{1:T}, s_{1:T}, \pi) = P(\pi) \prod_{t=1}^{T} P(o_t|s_t) P(s_t|s_{t-1}, \pi),$$

where π is a policy (sequence of actions) and $P(\pi) \propto \exp(-\mathbf{G}(\pi))$ with $\mathbf{G}(\pi)$ the **expected free energy**.

The expected free energy decomposes as:

$$\mathbf{G}(\pi) = \underbrace{\sum_{t=\tau}^{T} \mathbb{E}_{Q(o_t, s_t|\pi)} [\log Q(s_t|\pi) - \log P(s_t|\pi)]}_{\text{Expected epistemic value (information gain)}} + \underbrace{\sum_{t} \mathbb{E}_{Q(s_t|\pi)} [-\log P(o_t)]}_{\text{Expected extrinsic value (pragmatic)}},$$

where $P(o_t)$ encodes prior preferences over observations (the "goal"). The first term drives the agent toward states that reduce uncertainty about hidden causes (curiosity, information-seeking); the second toward states that match preferred outcomes (goal-directedness). Optimal policies under Active Inference thus automatically balance exploration and exploitation.

The connection to classical POMDP optimal control is: if $P(o) = \pi_0(o)$ encodes a reward function via $R(o) = -\log P(o)$, then minimising the extrinsic expected free energy is equivalent to maximising expected reward. Active Inference subsumes reward-maximisation as a special case but extends it with the epistemic drive, which has no counterpart in classical POMDP theory.

0.73.3 9b.3 The Free Energy of the World

The most ambitious claim of the Free-Energy Principle is that the dynamics of any self-organising system at non-equilibrium steady state can be described as free-energy minimisation. The mathematical formulation uses **stochastic differential equations** on the state space of the system.

A biological system at NESS occupies a submanifold $\mathcal{M}$ of its total state space, characterised by a non-equilibrium steady-state density $p^*(x)$. The **surprisal** $-\log p^*(x)$ plays the role of the potential function.

Friston's (2019) formulation models the system's dynamics as:

$$\dot{x} = (f + R)\nabla \ln p^*(x) + \nabla \cdot R,$$

where f is the antisymmetric (solenoidal) flow on the NESS manifold and R is the diffusion tensor. This can be decomposed into a gradient flow on $-\log p^*$ and a divergence-free component. The gradient flow component is precisely free-energy descent; the solenoidal component corresponds to the perpetual motion of the system's attracting manifold.

This formulation is mathematically demanding and physically controversial. Its strength is that it provides a single unifying principle for all dynamics at NESS; its weakness is that the free-energy decomposition is not unique, and the claim that the system is "minimising free energy" in a purposive sense requires additional assumptions beyond the mathematical structure.

0.74 9c. Markov Blankets, Self-Organisation, and Life

0.74.1 9c.1 The Markov Blanket Formalism

The Free-Energy Principle rests on the concept of the **Markov blanket**: the minimal set of states that "shield" a system from its environment, mediating all causal interactions between inside and outside.

Definition 0.74.1 (Markov Blanket)**.** For a system with state variables $\mathbf{x}$, the **Markov blanket** $\mathbf{b}$ of an internal set $\mathbf{i} \subseteq \mathbf{x}$ is the set of external states that render $\mathbf{i}$ conditionally independent of all remaining external states:

$$\mathbf{i} \perp\!\!\!\perp (\mathbf{x} \setminus \mathbf{b} \setminus \mathbf{i}) \mid \mathbf{b}.$$

In a directed graphical model (Bayesian network), the Markov blanket of a node i consists of its parents, its children, and the other parents of its children (the "co-parents"). In the context of biological systems, Friston identifies four types of states: **active states a** (motor efferents), **sensory states s** (sensory afferents), **internal states $\boldsymbol{\mu}$** (interoceptive), and **external states $\boldsymbol{\eta}$** (the environment). The Markov blanket is $\mathbf{b} = (\mathbf{a}, \mathbf{s})$.

The key claim of the FEP is that if a system maintains a Markov blanket over time (i.e., if the blanket persists as a partition), then the system's internal states must encode an implicit model of the external states — the internal states are a (approximately minimal sufficient) statistic for the external states given the blanket. This is the sense in which living systems "model" their environment.

Formally, the density of internal states $p(\boldsymbol{\mu}|\mathbf{b})$ conditional on blanket states encodes beliefs about external states. Since $\boldsymbol{\mu} \perp\!\!\!\perp \boldsymbol{\eta}|\mathbf{b}$, we have $p(\boldsymbol{\mu}, \boldsymbol{\eta}|\mathbf{b}) = p(\boldsymbol{\mu}|\mathbf{b})\, p(\boldsymbol{\eta}|\mathbf{b})$: internal and external states are conditionally independent given the blanket. But the marginals $p(\boldsymbol{\mu}|\mathbf{b})$ and $p(\boldsymbol{\eta}|\mathbf{b})$ share information about the blanket; the system's "beliefs" about external states are the mapping $\boldsymbol{\mu} \mapsto p(\boldsymbol{\eta}|\boldsymbol{\mu}, \mathbf{b})$ — the parametric family of distributions over external states indexed by internal states.

0.74.2 9c.2 Active Inference and Reinforcement Learning

The formal connection between Active Inference and **reinforcement learning** (RL) clarifies both frameworks. In the standard RL setting (MDPs), an agent maximises expected cumulative reward $\mathbb{E}[\sum_t \gamma^t R(s_t, a_t)]$.

In Active Inference, the agent minimises expected free energy:

$$\mathbf{G}(\pi) = \sum_t \underbrace{\mathbb{E}_{Q(o_t|\pi)}[-\log P(o_t)]}_{\text{Pragmatic (reward)}} + \underbrace{\mathrm{I}(o_t; s_t|\pi)}_{\text{Epistemic (exploration)}} \quad .$$

Setting the log-prior preference $-\log P(o)$ equal to the negative reward $-R(o)$ and ignoring the epistemic term, Active Inference reduces to expected reward maximisation: $\mathbf{G} \approx -\mathbb{E}[R(o)]$. The epistemic term is the additional contribution of Active Inference: it automatically drives **information-seeking behaviour** (exploration) even in the absence of external reward.

This provides an information-theoretic resolution of the **exploration-exploitation dilemma**: in classical RL, exploration is added *ad hoc* (epsilon-greedy, UCB, Thompson sampling). In Active Inference, exploration emerges from the minimisation of the expected free energy's epistemic component, which rewards policies that resolve uncertainty about hidden states. The balance between exploration and exploitation is set by the relative magnitude of the epistemic and pragmatic terms, which is determined by the agent's precision (confidence) in its generative model.

The **maximum entropy RL** framework (Ziebart 2010; Haarnoja et al., 2018) provides a complementary connection. In MaxEnt RL, the agent maximises:

$$J(\pi) = \sum_t \mathbb{E}[R(s_t, a_t) + \alpha\, \mathrm{H}(\pi(\cdot|s_t))],$$

where the entropy regulariser $\alpha\, \mathrm{H}(\pi)$ prevents the policy from becoming too deterministic. This objective corresponds to Active Inference with flat epistemic value and the pragmatic value set by the reward: the entropy term in the policy mirrors the entropy of the recognition distribution in variational inference.

Section V

Section V — Integrated Information Theory

0.75 10. The Hard Problem and the Mathematical Turn

Consciousness is the most recalcitrant problem in science. It is not merely difficult in the way that protein folding is difficult or climate prediction is difficult — complex, computationally intensive, but ultimately tractable within the existing framework of physical science. It is difficult in a prior way: it is not clear what a successful theory would look like, or what it would mean to explain why there is subjective experience at all rather than merely information processing in the dark.

David Chalmers called this the **Hard Problem**: why does the physical processing of information give rise to phenomenal experience? Why is there something it is like to be a functioning brain? This question has troubled philosophy for centuries and troubled cognitive science since its inception.

Giulio Tononi's **Integrated Information Theory** (IIT) is the most mathematically developed attempt to meet the Hard Problem on its own terms. It does not try to reduce consciousness to computation or explain it away. Instead, it proposes a formal identification: consciousness is identical to integrated information, and integrated information can be defined with mathematical precision for any system. The quantity Φ (phi) measures the amount of integrated information in a system, and it is the theory's central claim that Φ equals the amount of consciousness.

0.76 11. Integrated Information: The Formal Structure

0.76.1 11.1 Cause-Effect Structures

IIT is built on the concept of **intrinsic causal power**: the capacity of a system to exert causal influence on its own states, above and beyond the causal influence of its parts in isolation.

Consider a system of n binary variables $X = (X_1, \ldots, X_n)$ with a **transition probability matrix** (TPM) specifying $p(X^{(t+1)} = x^{(t+1)} \mid X^{(t)} = x^{(t)})$ for all state transitions. The system's current state $x^{(t)}$ is both a cause of future states and an effect of past states. The **cause-effect structure** of a state $x^{(t)}$ is the complete specification of how that state specifies its causes and effects.

For a mechanism $M \subseteq X$ (a subset of variables) with purview $Z \subseteq X$ (the subset of variables over which its causal power is evaluated), the **cause repertoire** is the distribution over past states of Z that are specifically selected by the current state $x_M^{(t)}$ of M:

$$p_{\text{cause}}(z^{(t-1)} \mid x_M^{(t)}) \propto p(x_M^{(t)} \mid z^{(t-1)})\, p(z^{(t-1)}),$$

where $p(z^{(t-1)})$ is the maximum entropy distribution over Z's past states (reflecting ignorance about unselected causes). Symmetrically, the **effect repertoire** is the distribution over future states of Z constrained by intervening on M:

$$p_{\text{effect}}(z^{(t+1)} \mid do(x_M^{(t)})) = \prod_{z_i \in Z} p(z_i^{(t+1)} \mid do(x_M^{(t)})).$$

The effect repertoire uses Pearl's do-calculus: we *intervene* to fix $X_M = x_M^{(t)}$ rather than merely observing, isolating the mechanism's genuine causal contribution from correlational structure.

0.76.2 11.2 Integrated Information Φ

The **integrated information** ϕ of a mechanism M with purview Z in state $x^{(t)}$ measures how much the mechanism's causal power *as a whole* exceeds the sum of its parts' causal power. It is the Earth Mover's Distance (Wasserstein-1 distance) between the actual cause-effect distribution of M and the product of the cause-effect distributions of the parts of M under the **minimum information partition** (MIP):

$$\phi(M, Z) = W_1(p_{\text{CE}}(z \mid x_M),\ p_{\text{MIP}})\,,$$

where p_{CE} is the cause-effect repertoire of the intact system and p_{MIP} is the cause-effect distribution after partitioning at the cut that minimises the loss of information (the minimum information partition). The Wasserstein-1 distance, rather than KL divergence, is used because it satisfies the requirement that the distance between distributions should reflect the work needed to transform one into the other, respecting the intrinsic geometry of the state space.

The system-level integrated information Φ is defined for the **complex**: the set of elements with maximal ϕ that is not part of a larger set with higher ϕ:

$$\Phi = \min_{\text{cut}} W_1\left(p_{\text{CE}}(x^{(t)}), p^{\text{cut}}_{\text{CE}}(x^{(t)})\right),$$

where the minimum is over all bipartitions of the system. A system with $\Phi = 0$ has no consciousness; the amount of consciousness is proportional to Φ.

0.76.3 11.3 Mathematical Properties of Φ

Φ has several mathematically distinctive properties.

Intrinsicality. Φ is defined from the system's own perspective, using the system's internal causal structure, not the perspective of an external observer. It is not a measure of information flow in any ordinary sense; it is a measure of causal integration.

Composition. A system's cause-effect structure is the union of the cause-effect structures of all mechanisms at all orders (single variables, pairs, triples, ...). Consciousness, on this view, is not a property of a single mechanism but of the complete multi-order causal structure.

Exclusion. A complex has definite borders (the maximal Φ set) and a definite temporal grain (the timescale at which Φ is maximal). This rules out overcounting: the same physical substrate does not count twice.

Computational hardness. Exact computation of Φ is #P-hard in the number of system elements: it requires searching over all bipartitions of the system, each requiring evaluation of an Earth Mover's Distance. For a system with n elements, the number of bipartitions is $2^{n-1} - 1$, growing exponentially. Realistic brain networks ($n \sim 10^{10}$) are far beyond exact computation; approximation methods and spectral surrogates (based on the eigenvalues of the connectivity matrix) are active areas of research.

Phenomenal axioms. IIT derives its mathematical structure from five phenomenological axioms: existence (experience exists), composition (each experience is structured), information (each experience is specific), integration (each experience is unified), and exclusion (each experience has definite borders). The mathematical formalism is claimed to be the unique formalisation of these axioms; the derivation

uses arguments from information geometry and order theory.

0.76.4 11.4 IIT and the Geometry of Qualia

One of IIT's most striking implications is a geometric account of qualitative properties of experience (**qualia**). Each experience corresponds to a point in **qualia space**: a high-dimensional geometric structure defined by the complete cause-effect structure of the complex. Two experiences are qualitatively similar if and only if they are geometrically close in qualia space, with similarity measured by the Wasserstein distance between their cause-effect structures.

This is a genuinely novel mathematical hypothesis: not that qualia are reducible to functional properties, but that their qualitative character — the redness of red, the sharpness of pain — is identical to geometric position in an intrinsic causal space. Whether this identification is correct is an empirical and philosophical question. But the mathematical structure is precise, and precision is the precondition of testability.

0.77 9d. Message Passing and the Neural Implementation of Inference

0.77.1 9d.1 Belief Propagation as Predictive Coding

The predictive coding architecture of Section 9.3 can be reinterpreted as a form of **belief propagation** on the hierarchical generative model. This connection, established by Friston and colleagues (2005), clarifies why the brain's cortical hierarchy might implement approximately correct Bayesian inference.

Consider a three-level hierarchy with states $\mu^{(1)}, \mu^{(2)}, \mu^{(3)}$ and observations $x^{(1)}$:

$$x^{(1)} = g^{(1)}(\mu^{(2)}) + z^{(1)}, \quad \mu^{(2)} = g^{(2)}(\mu^{(3)}) + z^{(2)},$$

with $z^{(l)} \sim \mathcal{N}(0, \Sigma^{(l)})$. The joint model is:

$$p(x^{(1)}, \mu^{(2)}, \mu^{(3)}) = p(x^{(1)}|\mu^{(2)})p(\mu^{(2)}|\mu^{(3)})p(\mu^{(3)}).$$

The variational free energy with mean-field approximate posterior $q(\mu^{(2)}, \mu^{(3)}) = q(\mu^{(2)})q(\mu^{(3)})$ is:

$$\mathcal{F} = -\mathbb{E}_q[\log p(x^{(1)}, \mu^{(2)}, \mu^{(3)})] + \mathbb{E}_q[\log q(\mu^{(2)})q(\mu^{(3)})].$$

Taking the gradient with respect to $\mu^{(2)}$ (the representation at level 2):

$$\frac{\partial \mathcal{F}}{\partial \mu^{(2)}} = (\Sigma^{(1)})^{-1}(g^{(1)}(\mu^{(2)}) - x^{(1)}) \cdot J_{g^{(1)}} - (\Sigma^{(2)})^{-1}(\mu^{(3)} - \mu^{(2)}).$$

Setting to zero and interpreting:

- The first term is a **bottom-up prediction error**: the precision-weighted residual between the prediction $g^{(1)}(\mu^{(2)})$ and the observation $x^{(1)}$, back-propagated through the Jacobian $J_{g^{(1)}}$.
- The second term is a **top-down prediction error**: the precision-weighted residual between the prediction from above $\mu^{(3)}$ and the current state $\mu^{(2)}$.

The fixed-point condition sets bottom-up and top-down prediction errors equal, balancing sensory evidence against prior expectations. Gradient descent on $\mathcal{F}$ over $\mu^{(2)}$ produces the dynamics of predictive

coding, with prediction error signals flowing upward (from level 1 to level 2) and predictions flowing downward (from level 3 to level 2).

This is precisely belief propagation on a chain graphical model: the messages are the prediction errors (bottom-up messages) and the predicted states (top-down messages). The mean-field approximation (factorising the posterior over levels) corresponds to the assumption that levels are conditionally independent given observations — approximately valid when the hierarchy is deep and top-down connections are weak.

0.77.2 9d.2 Precision and Attention

A distinctive feature of the predictive coding framework is the role of **precision** (inverse variance) in modulating the influence of prediction errors. In the gradient descent equations, each prediction error is weighted by its precision: high-precision (low-uncertainty) signals dominate over low-precision (high-uncertainty) signals.

Friston (2009) identified this precision-weighting as the mathematical implementation of **attention**: a high-precision sensory channel receives more weight in updating beliefs, which is exactly the phenomenology of attentional selection. Formally, attention modulates the effective precision $\tilde{\Pi}^{(l)} = \lambda^{(l)}\Pi^{(l)}$ where $\lambda^{(l)}$ is an attentional gain parameter. The optimised attentional gain satisfies:

$$\lambda^{(l)*} = \arg\min_{\lambda} \mathcal{F}(\lambda), \qquad \frac{\partial \mathcal{F}}{\partial \lambda^{(l)}} = \frac{1}{2}\Big((\epsilon^{(l)})^\top \Pi^{(l)} \epsilon^{(l)} - \mathrm{tr}(\Pi^{(l)}) \Big),$$

where $\epsilon^{(l)}$ is the prediction error at level l. Attention increases when the prediction error $\|\epsilon^{(l)}\|^2$ exceeds the expected value $\mathrm{tr}(\Pi^{(l)-1})$ — i.e., when sensory inputs are more surprising than expected under the current model. This is a formal account of the **mismatch negativity** (MMN) and other attention-capturing effects in cognitive neuroscience.

The precision-weighting also connects to **dopaminergic signalling**: dopamine neurons in the basal ganglia are thought to encode reward prediction errors (the difference between actual and expected reward). In the Active Inference framework, these correspond to the precision-weighted prediction errors in the action-selection layer of the hierarchy, connecting the biochemistry of dopamine to the mathematics of variational inference.

0.78 9e. Variational Message Passing and Free-Energy Minimisation in Neural Circuits

0.78.1 9e.1 Variational Message Passing

Variational Message Passing (VMP; Winn and Bishop 2005) is an algorithm that performs variational Bayes in a message-passing framework, making it possible to implement CAVI as local computations on a graphical model.

For a model $p(\mathbf{x}, \mathbf{z})$ with a mean-field approximation $q(\mathbf{z}) = \prod_i q_i(z_i)$, the optimal q_i satisfies (Section 5.2):

$$\log q_i^*(z_i) = \mathbb{E}_{j \neq i}[\log p(\mathbf{x}, \mathbf{z})] + \text{const.}$$

The expectation $\mathbb{E}_{j \neq i}[\log p(\mathbf{x}, \mathbf{z})]$ involves only the factors $p_a(\mathbf{x}_a)$ whose scope $\mathbf{x}_a$ includes z_i. In VMP, this expectation is computed by passing messages from the surrounding factors to node i:

Message from factor f_a to variable z_i:

$$\mu_{f_a \to z_i}(z_i) = \exp\big(\mathbb{E}_{q_{-i}}[\log f_a(\mathbf{x}_a)]\big) = \exp\left(\int \log f_a(\mathbf{x}_a) \prod_{j \in \text{scope}(f_a) \setminus i} q_j(z_j)\, dz_j \right).$$

Belief update for z_i:

$$q_i(z_i) \propto \prod_{a \in \mathcal{N}(z_i)} \mu_{f_a \to z_i}(z_i).$$

For exponential family models, both the messages and the beliefs are in exponential form, and the message computation reduces to arithmetic on natural parameters. VMP is thus applicable at scale: each update is $O(\text{scope size})$, and the algorithm converges to the mean-field optimum as quickly as CAVI.

The neural implementation of VMP is the predictive coding circuit: the message from a factor (a neural population encoding the generative model's CPD) to a variable (a neural population encoding the hidden state) is the precision-weighted prediction, and the belief update is the gradient descent on the free energy. The cortical hierarchy is a graphical model; predictive coding is VMP on that graphical model.

0.78.2 9e.2 Spike-and-Slab Priors and Synaptic Plasticity

Spike-and-slab priors provide a Bayesian model of synaptic sparsity: the prior over a synaptic weight w is a mixture of a point mass at zero ("spike") and a broad continuous distribution ("slab"):

$$p(w) = (1 - \pi)\delta_0(w) + \pi \mathcal{N}(w; 0, \sigma^2).$$

The parameter π controls the sparsity: small π means most synapses are zero. In a variational Bayesian framework, the posterior $q(w)$ approximates the posterior, and the CAVI update for w alternates between:

- Updating the "inclusion probability" $q(\gamma = 1) = \tilde{\pi}$ (the posterior probability that $w \neq 0$).
- Updating the conditional distribution $q(w|\gamma = 1) = \mathcal{N}(w; \tilde{\mu}, \tilde{\sigma}^2)$ (the posterior weight given inclusion).

The ELBO for the spike-and-slab model factorises over synapses and can be maximised synapse by synapse.

The biological interpretation: synaptic plasticity is approximated CAVI on the spike-and-slab model. The synaptic weight w encodes the posterior mean $\tilde{\mu}$, the synaptic reliability encodes $1/\tilde{\sigma}^2$, and the probability of synaptic transmission (release probability) encodes $\tilde{\pi}$. Learning (CAVI) updates these three quantities based on the prediction errors at the synapse. This provides a normative account of long-term potentiation (updating $\tilde{\mu}$), synaptic scaling (updating $\tilde{\sigma}^2$), and synaptogenesis/pruning (updating $\tilde{\pi}$).

0.79 11b. Computational Complexity of Φ and Approximations

0.79.1 11b.1 The #P-Hardness of Φ

Exact computation of Φ is computationally intractable. To see why, note that evaluating Φ requires:

1. Computing cause and effect repertoires for all $2^n - 1$ non-empty subsets (mechanisms) of the system.

2. For each mechanism, finding the minimum information partition (MIP) over all bipartitions, each requiring computation of a Wasserstein-1 distance.

3. Computing the system-level Φ as the minimum over all bipartitions.

The number of bipartitions of an n-element set is $2^{n-1} - 1$. Computing Wasserstein-1 distance between two distributions over 2^n states requires solving a linear program in $O(4^n)$ variables. The total computational cost is exponential in both steps, placing the exact computation of Φ in the #P complexity class (counting solutions to NP-hard problems), strictly harder than NP.

For a brain with $n \sim 10^{11}$ neurons, exact Φ computation is astronomically beyond any conceivable computational resource. Several approximation strategies have been proposed:

Spectral Φ. Balduzzi and Tononi (2008) proposed Φ^E, computed from the spectrum of the effective information matrix, at $O(n^3)$ cost. This approximation is fast but may not capture the full cause-effect structure.

Φ^* (IIT 3.0 simplified). Oizumi et al. (2016) introduced approximations that reduce to principal component analysis of the system's covariance structure for linear Gaussian systems.

Dynamical independence measures. Barrett and Seth (2011) proposed measuring Φ via the log-ratio of the joint spectral entropy to the sum of marginal spectral entropies, computable from neural time series data using Granger causality methods.

0.79.2 11b.2 Empirical Tests and Predictions

Despite computational intractability, IIT makes specific empirical predictions that have been partially tested:

Consciousness and cortical integration. IIT predicts that the posterior cortex (known to have dense recurrent connectivity) should have higher Φ than the cerebellum (feedforward architecture, lower integration). This is consistent with the clinical observation that cerebellar damage causes movement disorders but not loss of consciousness, while posterior cortical damage can abolish specific conscious contents.

The Perturbational Complexity Index (PCI). Casali et al. (2013) developed PCI as an empirical proxy for Φ: a TMS-EEG measure of the complexity and irreducibility of the brain's response to perturbation. $\text{PCI} = K(\text{response}) / \log_2(n_{\text{channels}} \times n_{\text{timepoints}})$, where K is (an approximation to) Kolmogorov complexity. PCI reliably distinguishes wakeful awareness from deep sleep, general anesthesia, and disorders of consciousness, providing the strongest current experimental support for IIT's predictions.

0.79.3 11b.3 Critiques and Competing Theories

Aaronson's critique. Scott Aaronson (2014) argued that IIT is committed to extreme conclusions: a simple feedforward circuit (a grid of XOR gates) with sufficiently high Φ would be conscious, while complex information-processing systems with low Φ (serial computers, feedforward networks) would not, regardless of their behavioural sophistication. He argued this constitutes a *reductio ad absurdum* of the theory.

IIT's response: the theory is about *intrinsic* causal structure, not external behaviour. A system that is behaviourally complex but internally feedforward has no consciousness precisely because it lacks intrinsic integration. Whether this is a reductio or a prediction depends on prior intuitions about consciousness.

Global Workspace Theory (GWT). Baars and Dehaene's Global Workspace Theory locates consciousness in the broadcast of information from a central **global workspace** to specialised modules. Mathematically, GWT can be formalised as a bottleneck architecture: consciousness arises when a limited-capacity high-bandwidth communication channel (the workspace) broadcasts a selected representation to all cognitive modules. Unlike IIT, GWT is architecturally specified rather than measure-theoretically defined.

Predictive processing theories. Clark, Friston, and Hohwy argue that consciousness is an aspect of the brain's predictive model-building: a representation becomes conscious when it attains sufficient precision weighting to drive action and influence global predictions. This view is consistent with Active Inference (Section IV) but does not identify a specific scalar measure of consciousness.

0.80 11c. IIT 4.0: Refined Axioms and Updated Formalism

0.80.1 11c.1 From IIT 3.0 to IIT 4.0

Tononi, Albantakis, and collaborators released IIT 4.0 (2023) with several important revisions to the mathematical formalism. The core axioms and postulates were refined to avoid counterintuitive edge cases in IIT 3.0, and the measure of integrated information was updated to resolve inconsistencies in the original Wasserstein-based Φ.

The five axioms (unchanged in spirit, refined in statement):

1. **Existence:** Consciousness exists; it is the most certain thing there is. Any theory of consciousness must take its existence as an incontrovertible datum.
2. **Composition:** Consciousness is structured: every experience is composed of phenomenal distinctions. The structure of experience is rich, not a homogeneous whole.
3. **Information:** Every experience is specific: it is this particular experience rather than any other. The identity of an experience is its difference from all others.
4. **Integration:** Every experience is unified: it cannot be decomposed into independent parts. The parts of an experience are dependent on the whole.
5. **Exclusion:** Every experience has definite borders: it includes some aspects of the world and excludes others. No experience is all-encompassing.

The five postulates translate axioms into mathematical requirements on causal mechanisms:

1. **Intrinsic existence:** A cause-effect structure has intrinsic causal power (not merely correlational structure).
2. **Composition:** Subsets of mechanisms jointly specify cause-effect structures over subsets of variables.
3. **Information:** A mechanism's causal power is measured by the difference between the causal repertoire it specifies and the unconstrained (maximum entropy) repertoire.

4. **Integration:** A cause-effect structure is irreducible: it cannot be fully accounted for by the sum of its parts.

5. **Exclusion:** Of all possible cause-effect structures over a given physical substrate, only the one with maximum intrinsic causal power is the system's actual cause-effect structure.

0.80.2 11c.2 A Worked Example: The XOR Gate

Consider the simplest non-trivial system: three binary units A, B, C where $C^{(t+1)} = A^{(t)} \oplus B^{(t)}$ (XOR) and $A^{(t+1)} = C^{(t)}$, $B^{(t+1)} = C^{(t)}$ (copy). The state space is $\{0,1\}^3$; the TPM specifies all $8 \to 8$ transitions.

For state $ABC = 110$ ($A = 1, B = 1, C = 0$):

Mechanism $\{C\}$ with purview $\{A, B\}$. The cause repertoire: what past states of $\{A, B\}$ are most likely given $C = 0$? Using the TPM backwards: $C = 0$ is produced by $AB \in \{00, 11\}$ (XOR outputs 0 when both inputs are equal). So $p_{\text{cause}}(A^{(t-1)} = 0, B^{(t-1)} = 0 | C = 0) = p_{\text{cause}}(A = 1, B = 1 | C = 0) = 1/2$, $p_{\text{cause}}(AB \in \{01, 10\} | C = 0) = 0$.

The MIP for this mechanism partitions $\{A, B\}$ into $\{A\}$ and $\{B\}$. Under the partition, the cause repertoire factorises as:

$$p_{\text{MIP}}(a, b | C = 0) = p(a | C = 0) \cdot p(b | C = 0) = \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4} \text{ for all } a, b.$$

(Under the partition, each variable is independent.)

The ϕ of mechanism $\{C\}$ is:

$$\phi = W_1(p_{\text{cause}}, p_{\text{MIP}}) = W_1 \left(\begin{bmatrix} 1/2 \\ 0 \\ 0 \\ 1/2 \end{bmatrix}, \begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{bmatrix} \right).$$

Computing the Wasserstein-1 distance (Earth Mover's Distance) on $\{00, 01, 10, 11\}$ with Hamming distance: the optimal transport plan moves 1/4 of mass from $(0,0)$ to $(0,1)$ at cost 1, and 1/4 of mass from $(1,1)$ to $(1,0)$ at cost 1, giving $W_1 = 1/4 \cdot 1 + 1/4 \cdot 1 = 1/2$. Thus $\phi(\{C\}, \{A, B\}) = 1/2$.

This calculation illustrates the core idea: the mechanism $\{C\}$ "constrains" the cause distribution in a way that cannot be decomposed into independent constraints from individual variables, yielding $\phi > 0$.

0.80.3 11c.3 Phenomenal Structure and Qualia Space

IIT's most striking claim is that the **geometry** of the cause-effect structure corresponds to the **phenomenal character** of experience. This is operationalised as follows.

For a complex in state x, the complete cause-effect structure (CES) is the collection of all cause-effect repertoires for all mechanisms and all their purviews. Two experiences are phenomenally identical iff their CES are identical; they are qualitatively similar to the extent that their CES are geometrically close in the space of probability distributions (measured by Wasserstein distance).

The **shape of experience** is thus determined by the geometry of the CES. IIT predicts:

- **Colour space:** The three-dimensional structure of human colour experience (hue, saturation, brightness) should correspond to the three-dimensional structure of the cause-effect geometry of the visual cortex's colour-processing circuits. Preliminary calculations on cone-opponent circuits yield the correct qualitative structure.
- **Pain and pleasure:** The valence (positive/negative) of an experience should correspond to the sign of the difference between the system's actual cause-effect structure and the structure it "prefers" (its attractor state). Systems driven toward their attractors have positive valence; systems driven away have negative valence.
- **Time and space:** The phenomenal experience of temporal succession and spatial extent should correspond to the causal-temporal and spatial structure of the CES's evolution and topology.

These predictions are currently in the speculative-but-testable range: they are more precise than any other theory of consciousness, but the measurements required to test them (computing Φ for neural circuits responding to different stimuli) are computationally intractable at scale.

Section *VI*

Section VI — Information Theory of Mind

0.81 12. Shannon Entropy and Cognitive Uncertainty

Claude Shannon's 1948 paper established information theory as a mathematical discipline. Its central object is **entropy**: a measure of the uncertainty in a random variable, or equivalently, the average surprise of an outcome.

Definition 0.81.1 (Shannon Entropy)**.** The entropy of a discrete random variable X with pmf p is

$$\mathrm{H}(X) = -\sum_{x} p(x) \log_2 p(x)$$

in bits, with the convention $0 \log 0 = 0$.

Entropy is non-negative, equals zero iff X is deterministic, and is maximised (equal to $\log_2 |\mathcal{X}|$ bits) when X is uniformly distributed. For a Bernoulli random variable with $p(\mathrm{H}) = \theta$, the entropy is $\mathrm{H}(\theta) = -\theta \log_2 \theta - (1-\theta)\log_2(1-\theta)$, which is zero at $\theta = 0$ and $\theta = 1$ (no uncertainty) and maximal at $\theta = 1/2$ (maximum uncertainty).

Mutual information between two random variables X and Y measures their statistical dependence:

$$\mathrm{I}(X;Y) = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} = \mathrm{H}(X) - \mathrm{H}(X \mid Y) = \mathrm{H}(Y) - \mathrm{H}(Y \mid X).$$

$\mathrm{I}(X;Y) \geq 0$ with equality iff X and Y are independent. In cognitive terms, mutual information measures how much knowing Y reduces uncertainty about X, and vice versa. It is the fundamental measure of information transfer in neural circuits, sensory channels, and communication between cognitive systems.

0.82 13. Algorithmic Information Theory and Kolmogorov Complexity

Shannon entropy measures average information over a distribution. **Kolmogorov complexity** $K(x)$ measures the information content of a *specific* string x: the length of the shortest program that generates x on a fixed universal Turing machine.

Formally, $K(x) = \min\{|\rho| : U(\rho) = x\}$ where U is a universal prefix-free Turing machine and $|\rho|$ is the length (in bits) of program ρ. Kolmogorov complexity is not computable: no algorithm can compute $K(x)$ for all x, because this would solve the halting problem. But it is **approximable from above**: any compressor provides an upper bound.

The **Minimum Description Length** (MDL) principle, due to Rissanen, translates Kolmogorov complexity into a practical model selection criterion: given data D and a model class $\mathcal{M}$, select the model $M^* \in \mathcal{M}$ that minimises the total description length of the model plus the data encoded under the model:

$$M^* = \arg\min_{M \in \mathcal{M}} \left[L(M) + L(D \mid M) \right],$$

where $L(M)$ is the description length of the model and $L(D \mid M) = -\log_2 p(D \mid M)$ is the description length of the data given the model. MDL is a formalisation of Occam's Razor: prefer the simplest model that explains the data. It is asymptotically equivalent to Bayesian model selection with a universal prior.

For cognitive science, Kolmogorov complexity and MDL provide a theory of what it means to find a good

explanation. An agent that seeks short descriptions of its experience is an agent that prefers simple, general hypotheses to complex, specific ones. This preference is not a heuristic but an informational necessity: short descriptions are the ones that compress most efficiently, transmit most reliably, and generalise most robustly.

0.83 12e. Information, Entropy, and the Physics of Cognition

0.83.1 12e.1 Maxwell's Demon and Landauer's Principle

Maxwell's demon (Maxwell 1867) is a thought experiment that threatened to violate the second law of thermodynamics using information. A demon operates a small door between two gas chambers: when a fast molecule approaches from the right, the demon opens the door and lets it through; when a slow molecule approaches from the left, the demon opens the door. Over time, the right chamber accumulates fast (hot) molecules and the left accumulates slow (cold) molecules, decreasing entropy without apparent work expenditure.

The resolution (Szilard 1929; Landauer 1961; Bennett 1982) is information-theoretic: the demon must *remember* the positions and speeds of molecules to make decisions. Each measurement stores one bit of information in the demon's memory. When the demon's memory is full, it must be **erased** to continue. By Landauer's principle:

Theorem 0.83.1 (Landauer's Principle)**.** Erasing one bit of information from a system in contact with a thermal reservoir at temperature T requires dissipating at least $k_B T \ln 2$ joules of energy.

The minimum energy cost of erasing one bit is $k_B T \ln 2 \approx 2.8 \times 10^{-21}$ J at room temperature. This energy appears as heat in the reservoir, restoring the entropy that the demon's measurements extracted. The second law is not violated; the demon simply converts information (order in its memory) to heat (disorder in the environment) during erasure.

Proof sketch. By the second law, the total entropy of system + demon + reservoir cannot decrease. A logically irreversible operation (many-to-one mapping, like erasing a bit: both 0 and 1 map to 0) must increase entropy elsewhere to compensate. The minimum entropy increase is one bit ($\ln 2$ nats), dissipated as heat $Q \geq k_B T \ln 2$.

Cognitive implications. If the brain performs computation, and computation (when irreversible) generates heat, then the brain's metabolic cost sets a lower bound on its computational irreversibility. A brain using ~ 20 W at $T = 310$ K can perform at most $Q/(k_B T \ln 2) \approx 20/(4.3 \times 10^{-21}) \approx 4.6 \times 10^{21}$ irreversible bit operations per second. This is a vast overestimate of actual neural computation, suggesting the brain is far from the Landauer limit — consistent with the view that the brain is not heat-limited but rather structure-limited (by connectivity, bandwidth, and precision).

0.83.2 12e.2 Free Energy and the Helmholtz Machine

The connection between thermodynamic free energy and variational free energy is not merely metaphorical. Helmholtz's free energy $F = U - TS$ (where U is internal energy, T temperature, S entropy) is minimised at thermodynamic equilibrium. The variational free energy $\mathcal{F}(q) = \mathbb{E}_q[E] - \mathrm{H}(q)$ (where $E = -\log p$ is the energy function) is minimised at the Bayesian posterior $q = p(\theta|D)$.

Both minimise the same structure: **expected energy minus entropy**. The variational free energy is the information-theoretic analogue of the Helmholtz free energy, with energy $= -\log$ prior$\times$likelihood and entropy $= H(q)$. The Bayesian posterior is the "equilibrium" state of an information-processing system.

The **Helmholtz machine** (Hinton and Zemel 1994) makes this analogy explicit: a two-pass neural network with a **generative model** (top-down, analogous to the thermodynamic system) and a **recognition model** (bottom-up, analogous to the demon measuring the system). The generative model minimises energy; the recognition model maximises entropy (of its beliefs). Together, they minimise the variational free energy, performing approximate Bayesian inference.

0.83.3 12e.3 Negentropy and the Free-Energy of Living Systems

Schrödinger (1944) famously argued that living systems maintain their organisation by feeding on **negentropy** (negative entropy) from the environment. Formally, a living system at non-equilibrium steady state has a stationary probability distribution $p^*(\mathbf{x})$ over its states, with entropy $H(p^*) < H_{\max}$ (strictly less than the maximum possible entropy). The difference $H_{\max} - H(p^*) \geq 0$ is the negentropy: the amount by which the living system is more ordered than a random system.

The Friston FEP formalises this: the living system minimises its variational free energy $\mathcal{F}(q, \mathbf{x}) = \mathrm{KL}(q(\tilde{\mu}) \| p(\tilde{\mu}|\tilde{s})) + \underbrace{(-\log p(\tilde{s}))}_{\text{surprisal}}$ where $\tilde{s}$ are sensory states and $\tilde{\mu}$ are internal states. Minimising $\mathcal{F}$ simultaneously minimises surprisal (the system avoids unlikely sensory states, i.e., states inconsistent with being alive) and optimises the recognition model (the system accurately infers the causes of its sensory states). The system's negentropy is maintained by this ongoing process of free-energy minimisation.

0.84 13f. Entropy Production and Cognitive Irreversibility

0.84.1 13f.1 Entropy Production in Stochastic Systems

The **entropy production rate** of a stochastic system is the rate at which thermodynamic entropy is produced in the environment due to the system's dynamics. For a Markov process on state space $\mathcal{S}$ with transition rates $W(s'|s)$, the entropy production rate is:

$$\dot{\sigma} = \sum_{s \neq s'} J(s, s') \ln \frac{W(s'|s)p(s)}{W(s|s')p(s')},$$

where $J(s, s') = W(s'|s)p(s) - W(s|s')p(s')$ is the probability current from s to s'. At equilibrium, $J = 0$ (detailed balance) and $\dot{\sigma} = 0$. At non-equilibrium steady state (NESS), $J \neq 0$ and $\dot{\sigma} > 0$: the system continuously produces entropy.

By the second law, $\dot{\sigma} \geq 0$, with equality only at equilibrium. The entropy production is related to the **time-reversal asymmetry**: $\dot{\sigma} = \lim_{T \to \infty} \frac{1}{T} \mathrm{KL}(P^+_{0:T} \| P^-_{0:T})$, where P^+ and P^- are the path measures of the forward and time-reversed processes. The KL divergence between forward and backward trajectories is the **thermodynamic cost of irreversibility**.

Cognitive irreversibility. Memory encoding is a paradigmatically irreversible process: information flows from the environment into the brain and is stored, but the reverse process (information spontaneously flowing from the brain back to the environment) does not occur in normal conditions. By Landauer's

principle, this irreversibility has a thermodynamic cost: each bit of information encoded in memory requires at least $k_B T \ln 2$ joules of energy dissipation.

0.84.2 13f.2 Mutual Information and Thermodynamic Efficiency

The **Szilard engine** (a Maxwell's demon that uses one bit of information to extract work $k_B T \ln 2$) establishes the connection between information and work:

$$\Delta W_{\text{extractable}} \leq k_B T \cdot \mathrm{I}(M; E),$$

where M is the demon's memory and E is the state of the environment (the gas). The maximum extractable work is $k_B T$ per bit of mutual information. This is the **information-work equivalence**: information about the environment can be converted to mechanical work, and vice versa.

For the brain, this means that neural representations of the environment carry potential **thermodynamic value**: the brain's model of the world could in principle be used to extract work from the environment (by performing optimal actions based on predictions). The quality of the brain's model — measured by mutual information $\mathrm{I}(M_{\text{brain}}; E)$ — determines the maximum extractable value. This provides a thermodynamic grounding for the utility of accurate mental models.

0.85 13e. Causal Information Theory

0.85.1 13e.1 Interventional Mutual Information

Standard Shannon mutual information $\mathrm{I}(X; Y)$ is a symmetric, purely observational quantity. **Causal mutual information** (CMI) is the information that variable X carries about Y when X is *intervened on*, rather than passively observed:

$$\mathrm{I}_{\text{causal}}(X \to Y) = \mathrm{I}_{do}(X; Y) = \int\int p(y|do(X = x)) \log \frac{p(y|do(X = x))}{p(y)} dy \, p_{\text{int}}(x) dx,$$

where $p(y|do(X = x))$ is the interventional distribution (from the do-calculus) and $p_{\text{int}}(x)$ is the intervention distribution (often uniform over the support of X).

CMI is not symmetric: $\mathrm{I}_{\text{causal}}(X \to Y) \neq \mathrm{I}_{\text{causal}}(Y \to X)$ in general. The difference $\mathrm{I}_{\text{causal}}(X \to Y) - \mathrm{I}_{\text{causal}}(Y \to X)$ is a measure of the **causal directionality**: positive values indicate that X causally influences Y more than vice versa.

Identifiability from observational data. The interventional distribution $p(y|do(x))$ is generally not identifiable from observational data without additional assumptions (the do-calculus). When the causal graph is known and there are no unobserved confounders, $p(y|do(x)) = p(y|x)$ (observational and interventional are equal), making CMI equal to standard MI. With confounders, adjusting by the backdoor criterion gives the correct interventional distribution.

0.85.2 13e.2 Transfer Entropy and Directed Information

Transfer entropy (TE; Schreiber 2000) is the information flow from time series X to Y that is not already present in Y's own past:

$$\text{TE}_{X\to Y} = \text{I}(X_{t-k:t-1}; Y_t | Y_{t-l:t-1}) = H(Y_t|Y_{t-l:t-1}) - H(Y_t|Y_{t-l:t-1}, X_{t-k:t-1}).$$

TE is the time-series analogue of Granger causality (Section 14f): it measures how much knowing the past of X reduces uncertainty about the current value of Y, beyond what is already predictable from Y's own past. Unlike Granger causality (which uses a linear model), TE is model-free.

Directed information (DI; Massey 1990; Permuter et al., 2008) is the total information flowing from $X^n = (X_1, \ldots, X_n)$ to Y^n in a causal manner:

$$\text{DI}(X^n \to Y^n) = \sum_{i=1}^{n} \text{I}(X^i; Y_i | Y^{i-1}) = \sum_{i=1}^{n} \text{I}(X_1^i; Y_i | Y_1^{i-1}),$$

where the conditioning on Y^{i-1} enforces causality (only the past of Y, not its future, is conditioned on). DI satisfies the **data processing inequality for causal systems**: for a Markov chain $X \to Y \to Z$ (causal), $\text{DI}(X \to Z) \leq \text{DI}(X \to Y)$.

For neural population activity, directed information provides a non-parametric measure of causal influence that is applicable to multi-unit recordings without specifying a dynamical model. The estimated DI from spike trains can be used to infer the functional connectivity graph of a neural circuit, with directed edges indicating significant causal information flow.

0.86 13d. Place Cells, Grid Cells, and the Neural Code for Space

0.86.1 13d.1 Place Cells and Rate Maps

Place cells (O'Keefe and Dostrovsky 1971) are neurons in the hippocampus that fire preferentially when the animal occupies a specific region of space (the **place field**). The firing rate of place cell i at position $\mathbf{r} = (x, y)$ is described by a tuning function:

$$f_i(\mathbf{r}) = A_i \exp\left(-\frac{|\mathbf{r} - \mathbf{r}_i^*|^2}{2\sigma_i^2} \right),$$

where $\mathbf{r}_i^*$ is the centre of the place field, σ_i its radius, and A_i the peak rate. Together, N place cells with uniformly distributed field centres tile the environment, providing a distributed, population-coded representation of position.

Maximum likelihood decoding. Given the spike counts $\mathbf{r} = (r_1, \ldots, r_N)$ in a short time window Δt, the decoded position is:

$$\hat{\mathbf{r}} = \arg\max_{\mathbf{r}} \log P(\mathbf{r}_{\text{spikes}} | \mathbf{r}) = \arg\max_{\mathbf{r}} \sum_i r_i \log f_i(\mathbf{r}) - \Delta t \sum_i f_i(\mathbf{r}).$$

For Poisson spiking with rates $f_i(\mathbf{r})\Delta t$, this is the maximum likelihood estimator. The Fisher information

for position decoding is:

$$\mathcal{I}(\mathbf{r}) = \sum_i \frac{1}{f_i(\mathbf{r})} \nabla_{\mathbf{r}} f_i(\mathbf{r}) (\nabla_{\mathbf{r}} f_i(\mathbf{r}))^\top = \sum_i \frac{(\mathbf{r} - \mathbf{r}_i^*)(\mathbf{r} - \mathbf{r}_i^*)^\top}{f_i(\mathbf{r}) \sigma_i^4}.$$

For a uniform population of place cells with equal parameters, $\mathcal{I}(\mathbf{r}) = (NA)/(2\pi\sigma^2) \cdot I_2$: position decoding accuracy scales linearly with the number of cells and inversely with the field size.

0.86.2 13d.2 Grid Cells and Hexagonal Grids

Grid cells (Moser, Moser, and Sargolini 2005) are neurons in the medial entorhinal cortex (MEC) that fire at multiple locations arranged in a regular hexagonal lattice:

$$f(\mathbf{r}) = A \left[1 + 2 \sum_{j=1}^{3} \cos(\mathbf{g}_j \cdot (\mathbf{r} - \boldsymbol{\phi})) \right]^+,$$

where $\mathbf{g}_j = \frac{4\pi}{\lambda\sqrt{3}}(\cos(j\pi/3 - \pi/6), \sin(j\pi/3 - \pi/6))$ are the three grid vectors with spacing λ and phase $\boldsymbol{\phi}$, and $[\cdot]^+$ is the rectification. The firing pattern is a **hexagonal lattice** with spacing λ, orientation set by the grid vectors.

The mathematical reason for hexagonal grids: the hexagonal lattice is the **densest packing** of circles in the plane (the densest way to cover the plane with place fields of equal radius). It also minimises the **quantisation error** for position estimation: given N firing fields over an area $\mathcal{A}$, the hexagonal arrangement minimises the average distance from any point to the nearest field centre, giving the best position resolution for a fixed number of cells.

Multiple scales and capacity. MEC contains grid cells at multiple spatial scales $\lambda_1 < \lambda_2 < \cdots < \lambda_K$ (in distinct **modules**). The combination of grid cells at different scales is analogous to a **Chinese Remainder Theorem** (CRT) encoding: a position in a large environment can be uniquely specified by the phase coordinates within each module, if the periods λ_k/λ_1 are coprime integers. The total number of uniquely representable positions is $\prod_k N_k$ (the product of positions per module), enabling exponentially large spatial memory with a linearly growing number of cells.

0.86.3 13d.3 Successor Representation and Predictive Maps

The **Successor Representation** (SR; Dayan 1993) provides an alternative to the place field model of hippocampal cells. Instead of encoding current position $\mathbf{r}$, the SR encodes the **expected future occupancy**: $M(\mathbf{r}, \mathbf{r}') = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \mathbf{1}[\mathbf{r}_t = \mathbf{r}'] | \mathbf{r}_0 = \mathbf{r}\right]$ — the discounted expected time the agent will spend at $\mathbf{r}'$ starting from $\mathbf{r}$.

The SR matrix M satisfies the **Bellman equation**: $M = (I - \gamma T)^{-1}$ where T is the transition matrix and γ is the temporal discount factor. The SR can be learned online by TD learning:

$$\Delta M(\mathbf{r}, \cdot) = \alpha(\mathbf{e}_{\mathbf{r}} + \gamma M(\mathbf{r}', \cdot) - M(\mathbf{r}, \cdot)),$$

where $\mathbf{e}_{\mathbf{r}}$ is a one-hot vector at the current position and $\mathbf{r}'$ is the next position. The SR provides a compact representation of the environment's predictive structure: features derived from the SR enable fast generalisation to new reward locations without re-learning (just re-weight the SR features by the new reward

vector w, giving value $V(\mathbf{r}) = M(\mathbf{r}, \cdot)^\top w$).

0.87 13c. Algorithmic Models of Cognitive Biases

0.87.1 13c.1 Representativeness and Kolmogorov Complexity

The **representativeness heuristic** (Tversky and Kahneman 1974) is the tendency to judge probabilities by how similar an event is to a prototype, ignoring base rates. For example, people judge "Linda is a bank teller and a feminist" more probable than "Linda is a bank teller" (the conjunction fallacy), because the conjunctive description is more *representative* of Linda's profile.

Chater and Vitányi (2003) provide an algorithmic formalisation: the probability of a string x under the simplicity prior is $P_M(x) \propto 2^{-K(x)}$ (Section 12c). The representativeness of x given prototype p is $2^{-K(x|p^*)}$ (how compressible is x given p?). The conjunction fallacy arises because:

$$K(\text{bank teller AND feminist}|\text{Linda's profile}) < K(\text{bank teller}|\text{Linda's profile}),$$

i.e., the conjunctive description is more compressible (shorter to describe) given Linda's profile. The agent using the simplicity-based representativeness heuristic judges the conjunction more probable, not because they ignore base rates, but because the *description length* model incorrectly equates compressibility with probability.

Base rate neglect. The true posterior probability is $P(\text{FC}|\text{profile}) = P(\text{profile}|\text{FC})P(\text{FC})/P(\text{profile})$. The representativeness heuristic ignores $P(\text{FC})$ (the base rate of feminist bank tellers) and reports $P(\text{profile}|\text{FC})$ alone. Algorithmically, this corresponds to using the conditional complexity $K(\text{profile}|\text{FC})$ without the complexity $K(\text{FC})$: the agent correctly estimates the description length of the profile given the hypothesis, but fails to account for the complexity of the hypothesis itself.

0.87.2 13c.2 The Availability Heuristic and Sampling Biases

The **availability heuristic**: people judge the probability of an event by the ease with which instances come to mind. Dramatic, salient, or recent events are judged as more probable because they are more easily "available" in memory.

Information-theoretically, availability corresponds to sampling from an approximate posterior that overweights salient (low K, easily describable) events. If the brain maintains a sample-based representation of its prior distribution over events (a **particle filter approximation** of the prior), and if saliency correlates with short description length, then salient events will be overrepresented in the particle set and thus judged as more probable.

The **availability bias** is then a systematic error in the particle approximation: the sampling distribution is not exactly the prior $P(\text{event})$ but a biased version $\tilde{P}(\text{event}) \propto P(\text{event}) \cdot \text{salience}(\text{event})$. Availability bias is a calibration error in the particle filter, not a fundamental failure of probabilistic reasoning.

0.87.3 13c.3 The Anchoring Effect as Insufficient Bayesian Updating

The **anchoring effect**: when asked to estimate an unknown quantity, people are biased toward an initial anchor value, even when the anchor is arbitrary. Classically, this is interpreted as a failure of rationality.

The Bayesian interpretation (Lieder et al., 2018): anchoring is the result of a limited number of MCMC updates. If the prior over the unknown quantity θ is initialised at the anchor θ_0, and the posterior is approximated by running T steps of MCMC starting from θ_0, then for small T the approximation will be biased toward θ_0 (the chain hasn't mixed). The anchoring effect is the Bayesian inference error due to insufficient MCMC mixing:

$$\hat{\theta}_T \approx \theta_0 + T \cdot \text{mixing rate} \cdot (\theta_{\text{posterior mean}} - \theta_0).$$

For large T (full mixing), $\hat{\theta}_T \to \theta_{\text{posterior mean}}$; for small T, $\hat{\theta}_T$ is biased toward θ_0. The magnitude of anchoring bias is therefore predicted by T (number of cognitive operations available) and the mixing time of the posterior (how far the posterior mean is from the anchor relative to the posterior spread).

0.88 13b. Information Theory and the Binding Problem

0.88.1 13b.1 The Binding Problem

The **binding problem** asks how the brain integrates information processed in spatially and temporally distributed neural populations into unified perceptual experiences. When you perceive a red apple, the colour red is processed in V4, the round shape in V2/V4, and the identity "apple" in the inferotemporal cortex — yet you perceive a unified object, not three independent features. What binds them?

Information-theoretic approaches model binding as **synergistic information**: information about the stimulus that is present in the joint distribution of neural responses but absent in any individual population's response. If $(R_1, R_2, \ldots, R_k)$ are the responses of k neural populations to stimulus S, the **partial information decomposition** (Williams and Beer, 2010) decomposes $\mathrm{I}(S; R_1, \ldots, R_k)$ into:

$$\mathrm{I}(S; R_1, \ldots, R_k) = \underbrace{\sum_i U_i}_{\text{Unique}} + \underbrace{\sum_{i<j} R_{ij}}_{\text{Redundant}} + \underbrace{Sy}_{\text{Synergistic}} .$$

The synergistic component Sy quantifies information that is present only in the joint distribution: information about S that cannot be recovered from any single population or any subset of populations, but requires reading all populations simultaneously. On this view, binding is the neural computation of synergistic information, and it is solved by neural circuits that create distributed representations with high joint mutual information.

Formal computation of partial information requires a non-negative information measure $I_\cap(S; R_1, R_2)$ (the redundant information between sources R_1, R_2 about S) satisfying specific axioms. The most widely used definition is $I_\cap(S; R_1, R_2) = \min(I(S; R_1), I(S; R_2))$, but this yields negative synergy in some cases. More principled alternatives based on the $I_{\min}$ **measure** or $I_\cap^{MMI}$ (minimum mutual information) remain active research.

0.88.2 13b.2 Rate-Distortion Theory and Efficient Coding

Rate-distortion theory formalises the trade-off between the fidelity of a representation and the number of bits required to store or transmit it. For a source X and a distortion function $d : \mathcal{X} \times \hat{\mathcal{X}} \to \mathbb{R}_+$, the

rate-distortion function is:

$$R(D) = \min_{p(\hat{x}|x):\mathbb{E}[d(X,\hat{X})]\leq D} \mathrm{I}(X;\hat{X}).$$

This is the minimum number of bits needed to represent X with expected distortion at most D. Shannon's rate-distortion theorem establishes that $R(D)$ is achievable (with codes of large block length) and that no code can achieve distortion D with fewer than $R(D)$ bits.

Barlow's **efficient coding hypothesis** proposes that the sensory nervous system has evolved to represent the environment with minimum redundancy: neural codes maximise the mutual information between the stimulus and the neural response, subject to constraints on neural firing rates (energy). This predicts that the distribution of neural responses should approach the distribution that maximises entropy given the firing rate constraint — a prediction verified in retinal ganglion cells (the center-surround receptive field is the whitening filter that removes spatial correlations in natural images) and simple cells in V1 (Gabor-like receptive fields are the basis functions of ICA applied to natural images).

0.88.3 13b.3 Kolmogorov Complexity and Solomonoff Induction

The **Solomonoff prior** $M(x) = \sum_{p:U(p)=x} 2^{-|p|}$ assigns to each binary string x a probability proportional to the sum of $2^{-|p|}$ over all programs p that output x on universal Turing machine U. This is a universal prior that assigns high probability to simple, compressible strings and low probability to complex ones.

Solomonoff's **universal induction** predicts the next symbol x_{n+1} given past data $x_{1:n}$ by:

$$P_M(x_{n+1} = 1 \mid x_{1:n}) = \frac{M(x_{1:n}1)}{M(x_{1:n}1) + M(x_{1:n}0)}.$$

The fundamental theorem of algorithmic information theory (Solomonoff 1964; Li and Vitányi 1997) states that universal induction converges to any computable generating distribution at least as fast as any other method: for any computable distribution μ:

$$\sum_{n=0}^{\infty} \mathbb{E}_\mu\big[(P_M(X_{n+1}|X_{1:n}) - \mu(X_{n+1}|X_{1:n}))^2\big] \leq \frac{\ln 2}{2} K(\mu),$$

where $K(\mu)$ is the Kolmogorov complexity of the generating distribution. Universal induction is optimal in the sense that it learns any computable environment with cumulative squared prediction error bounded by $K(\mu)/2$.

For cognitive science, Solomonoff induction is the limiting idealisation of Bayesian cognition: a cognitive agent with unlimited computational resources and a universal prior would converge to any computable environmental distribution at the fastest possible rate. Real brains approximate this ideal with neurally plausible priors (smooth functions, compositional structure) and approximate inference algorithms. The MDL principle operationalises this limit in practical model selection.

0.89 11i. Psychophysics of Consciousness: Mathematical Experiments

0.89.1 11i.1 The Threshold for Conscious Perception

Sensory threshold experiments measure the minimal stimulus strength required for conscious perception. For a signal of intensity s and detection criterion c (Section 20g):

$$P(\text{detect}|s) = \Phi\left(\frac{s - c}{\sigma_{\text{noise}}}\right),$$

the psychometric function. The threshold s_{50} (at which $P(\text{detect}) = 0.5$) is $s_{50} = c$. The slope $1/(\sigma_{\text{noise}}\sqrt{2\pi})$ at threshold reflects sensitivity.

Type-2 signal detection. The **Type-2 ROC** (Receiver Operating Characteristic) measures metacognitive accuracy (Section 20i): how well does the participant's confidence predict whether their response was correct? A perfect metacognitive system has Area Under the Curve (AUC) $= 1$; a system with no metacognitive ability has AUC $= 0.5$. The **meta-d'** signal-theoretic measure (Section 20i) is related to the AUC: meta-$d' = 2\Phi^{-1}(\text{AUC})$.

0.89.2 11i.2 Neural Correlates of Consciousness: Granger Cause and Effect

The search for the **neural correlates of consciousness** (NCC) requires distinguishing neural events that *constitute* conscious experience from those that merely *precede* or *follow* it. Granger causality (Section 14f) provides a tool for this: if area A Granger-causes the subjective report (button press) but not vice versa, it is a candidate NCC.

The **Granger causal structure** of consciousness: in masking experiments (where a near-threshold stimulus is followed by a mask), conscious perception correlates with late (> 300 ms) frontoparietal activity, while unconscious processing correlates with early (< 200 ms) occipital activity. Granger causality from frontal to parietal areas is specifically increased for conscious trials, suggesting that frontoparietal feedback is causally necessary for consciousness (consistent with GWT and predictive coding but inconsistent with local recurrent processing theories).

0.90 11h. Network Neuroscience and Integration Measures

0.90.1 11h.1 Functional Connectivity and Effective Connectivity

Functional connectivity (FC) in neuroimaging is the statistical dependence between neural time series: $\text{FC}_{ij} = \text{corr}(x_i(t), x_j(t))$ or equivalently the partial correlation, capturing direct dependencies after controlling for all other regions. FC is symmetric and observational.

Effective connectivity (EC) is the directed, causal influence of region i on region j: it requires a mechanistic model (DCM, structural equation model, or granger causality) and provides directed, potentially asymmetric connectivity estimates. EC is what IIT's cause-effect structure aims to capture: the actual causal architecture of the system.

Dynamic Causal Modelling (DCM; Friston et al., 2003) is the principled Bayesian approach to effective

connectivity. For a region $x_i(t)$ modulated by input $u(t)$:

$$\dot{x}_i = \sigma\left(A_i + \sum_j B_{ij}^{(u)} u(t)\right) x + Cu,$$

where A is the intrinsic connectivity matrix, $B^{(u)}$ is the input-modulated connectivity, and C is the input matrix. The Bayesian inversion of DCM estimates the posterior $p(A, B, C|\text{fMRI data})$ by variational Bayes, providing uncertainty estimates for all connectivity parameters.

0.90.2 11h.2 Graph-Theoretic Measures of Neural Integration

Graph theory provides complementary measures of neural integration that are more computationally tractable than Φ:

Global efficiency $E_{\text{glob}} = \frac{1}{N(N-1)} \sum_{i \neq j} \frac{1}{d_{ij}}$ (where d_{ij} is the shortest path length between regions i and j) measures how efficiently information can flow across the network. High global efficiency = short average paths = fast information integration.

Local efficiency $E_{\text{loc}} = \frac{1}{N} \sum_i E_{\text{glob}}(G_i)$ (where G_i is the subgraph of i's neighbours) measures the fault-tolerance of the network: how well information flows when a node is removed.

Modularity $Q = \sum_c [L_c/L - (d_c/2L)^2]$ (where L_c is the number of edges within community c and d_c is the total degree of community c) measures the degree to which the network is divided into distinct communities. High modularity = segregated, specialised processing; low modularity = highly integrated, generalised processing.

Small-worldness: a network is **small-world** if its clustering coefficient C greatly exceeds that of a random graph ($C \gg C_{\text{random}}$) while its characteristic path length L is comparable to a random graph ($L \approx L_{\text{random}}$). The small-world index $\sigma = (C/C_{\text{random}})/(L/L_{\text{random}}) \gg 1$. The human connectome is a small-world network: highly clustered (specialised modules) with short path lengths (efficient global communication). This topology is hypothesised to optimise the balance between local specialisation and global integration, with implications for Φ.

0.91 11g. Measure-Theoretic Foundations of Integrated Information

0.91.1 11g.1 Cause-Effect Repertoires as Probability Distributions

The **cause-effect repertoire** of a mechanism M in state m over purview P is a probability distribution over the states of P. For a discrete system with n binary units:

Cause repertoire $p_c(p|m, \text{MBL}_M^-)$: the probability distribution over past states of purview P that are compatible with the current state m of mechanism M. It is computed by Bayesian inversion: applying Bayes' theorem to the transition probability matrix (TPM) of the system.

For a TPM $T(s_{t+1}|s_t)$ specifying the probability of state s_{t+1} at time $t+1$ given state s_t at time t, the cause repertoire is:

$$p_c(p|m) \propto T(m^+|p^-) \cdot p_{\text{unconstrained}}(p),$$

where $m^+ =$ state of M at $t+1$, $p^- =$ state of purview P at t, and $p_{\text{unconstrained}}$ is the maximum entropy distribution over P's states. The $\propto$ normalises over all states of P.

Effect repertoire $p_e(p|m, \text{MBL}_M^+)$: the probability distribution over future states of purview P given the current state m of mechanism M. It is simply:

$$p_e(p|m) = \sum_{s^- : \text{state of } P^c} T(p^+|m^-, \{s^-\}) \cdot p_{\text{unconstrained}}(s^-),$$

where P^c are the units outside purview P and p^+ is the state of P at $t+1$.

0.91.2 11g.2 ϕ as an Optimal Transport Distance

The ϕ of a mechanism is the Wasserstein-1 distance between the intact cause-effect repertoire p and the partitioned cause-effect repertoire p_{MIP}:

$$\phi(M, P) = W_1(p(P|M), p_{\text{MIP}}(P|M)).$$

The Wasserstein-1 distance on a metric space $(\mathcal{P}, d_{\mathcal{P}})$ (the state space of P with Hamming distance):

$$W_1(p, q) = \inf_{\gamma \in \Gamma(p,q)} \sum_{(s,s') \in \mathcal{P}^2} d_{\mathcal{P}}(s, s')\, \gamma(s, s'),$$

where $\Gamma(p, q)$ is the set of couplings of p and q. For discrete distributions over $|\mathcal{P}| = 2^{|P|}$ states, this is a linear program with $|\mathcal{P}|^2$ variables and $2|\mathcal{P}|$ constraints.

The Hamming distance $d_{\mathcal{P}}(s, s') = |\{i : s_i \neq s'_i\}|$ (number of bit positions where s and s' differ) is the natural metric for binary state spaces. It reflects the idea that states differing in fewer units are more similar. Using Euclidean distance or other metrics would change the numerical values of ϕ and potentially the qualitative ordering of mechanisms.

0.91.3 11g.3 The Maximum of Small Φ: Why the System Matters

A key claim of IIT is that the level of a conscious system is determined by the **complex**: the set of units that maximises Φ. Not all units of the brain are part of the conscious complex; only those that form a maximally integrated cause-effect structure.

Formally, for a candidate system S:

$$\Phi(S) = W_1(\text{CES}(S), \text{CES}(\text{MIP}(S))),$$

where CES is the complete cause-effect structure (all cause-effect repertoires for all mechanisms and purviews) and MIP is the minimum information partition of the system. The complex is the candidate system with maximum Φ over all possible subsets of the physical substrate. Overlapping candidate systems are not allowed (the exclusion postulate): the complex "claims" the units that maximise Φ.

The maximum is over exponentially many subsets (2^N for an N-unit system), making it computationally intractable for large systems. For a 10-unit system, there are $2^{10} - 1 = 1023$ candidate systems, each requiring computation of Φ (which itself involves exponentially many bipartitions). For a brain with $N \sim 10^{11}$ neurons, exact computation is astronomically infeasible.

0.92 11f. Free Energy Minimisation Across Biological Scales

0.92.1 11f.1 Cellular Free Energy Minimisation

The FEP is not limited to neural systems. Friston (2013, 2019) argues that any self-organising biological system at non-equilibrium steady state can be characterised as minimising variational free energy. At the cellular level, a cell maintains its internal organisation against thermal dissipation by:

1. **Minimising surprisal**: maintaining biochemical concentrations within viable ranges (homeostasis). A cell that ventures into high-surprisal states (extreme pH, temperature, osmolarity) risks membrane rupture or protein denaturation.

2. **Active inference**: exerting active control over the intracellular environment (ion pumps, gene regulation, cytoskeletal dynamics) to minimise prediction errors between the current and expected intracellular state.

The mathematical formulation: let $\mathbf{m}$ be the internal state of the cell (metabolite concentrations, gene expression levels), $\mathbf{e}$ be the external environment, and $\mathbf{s}$ be the cell surface (sensors and actuators, the Markov blanket). The cell's dynamics are:

$$\dot{\mathbf{m}} = f_m(\mathbf{m}, \mathbf{s}), \qquad \dot{\mathbf{s}} = f_s(\mathbf{s}, \mathbf{m}, \mathbf{e}).$$

The cell minimises the free energy $\mathcal{F}(\mathbf{m}, \mathbf{s}) = -\log p(\mathbf{s}|\mathbf{m}) + \mathrm{KL}(q(\mathbf{e}|\mathbf{m}) \| p(\mathbf{e}|\mathbf{s}))$ by adjusting both its internal state $\mathbf{m}$ (perceptual inference) and its surface dynamics f_s (active inference). Gene regulatory networks, metabolic networks, and signalling cascades are the biological implementations of these inference computations.

0.92.2 11f.2 Developmental Neuroscience as Active Inference

The development of the nervous system involves the progressive refinement of neural circuits through a combination of genetically specified priors and activity-dependent learning. From the FEP perspective:

Synaptic pruning is the elimination of synapses that do not contribute to reducing prediction errors. Synapses that consistently carry redundant information (those whose activation is perfectly predictable from other inputs) are pruned; synapses that carry unique, irreducible information are strengthened. This is the developmental analogue of the sparse coding and dimensionality reduction principles of mature neural coding.

Critical periods are developmental windows during which the nervous system is maximally sensitive to sensory experience. Mathematically, a critical period is a period of high **synaptic plasticity** (high learning rate) followed by a period of low plasticity (consolidation). The onset and offset of critical periods are controlled by the balance of excitation and inhibition (the E/I ratio): high E/I $\rightarrow$ high plasticity; low E/I $\rightarrow$ consolidation. The FEP predicts that critical periods should coincide with periods of high prediction error (the nervous system is most uncertain about the world during early development, and this uncertainty drives the high plasticity that resolves it).

Mathematical model of critical period. Let $w(t)$ be the synaptic strength and $\epsilon(t)$ be the prediction

error at time t. A simple model of Hebbian-like plasticity with homeostatic regulation:

$$\dot{w} = \lambda(t)(\epsilon - \epsilon^*)w - \alpha w,$$

where $\lambda(t)$ is the time-varying learning rate (high during the critical period, low afterwards), ϵ^* is the target prediction error, and α is a weight decay term. The critical period is a period of high $\lambda(t)$; its close is a decrease in λ that "freezes" the synaptic weights at their current values.

0.93 11e. Empirical Measures of Consciousness

0.93.1 11e.1 The Perturbational Complexity Index

The **Perturbational Complexity Index** (PCI; Casali et al., 2013) provides an empirical proxy for Φ based on TMS-EEG measurements. The procedure:

1. Deliver a TMS pulse to the cortex (transcranial magnetic stimulation: a brief, focal magnetic field that excites cortical neurons).
2. Record the resulting EEG response for 300 ms post-stimulation.
3. Quantify the **complexity** of the response using a compression-based measure of Kolmogorov complexity: the response matrix (electrodes $\times$ time) is binarised using a source-modelling threshold, and the compressed size (using LZ76 algorithm) divided by the uncompressed size gives a normalised complexity estimate.

Formally:

$$\text{PCI} = \frac{K(s)}{\log_2(n_{\text{channels}} \times n_{\text{timepoints}})},$$

where $K(s)$ is the LZ76 compressed length of the binarised source matrix s and the denominator normalises by the maximum possible complexity (completely random binary matrix).

PCI reliably discriminates:

- **Wakefulness**: PCI > 0.44 in nearly all healthy subjects.
- **NREM deep sleep**: PCI < 0.31, consistent with IIT's prediction that sleep is associated with reduced Φ (the cortex breaks into locally active but globally uncommunicating regions).
- **General anaesthesia** (propofol, xenon, ketamine): PCI < 0.31 for propofol and xenon, but PCI near waking levels for ketamine (consistent with the dissociative states produced by ketamine, which preserve experience while disrupting perception and memory).
- **Minimally conscious state (MCS)**: PCI ≈ 0.44, distinguishable from vegetative state (PCI < 0.31) despite similar motor unresponsiveness.

The validation of PCI as a consciousness measure (rather than a complexity measure) requires the assumption that the TMS-EEG complexity reflects the underlying Φ of the cortical network. This assumption is not proven, but the empirical discrimination of conscious and unconscious states provides strong circumstantial support.

0.93.2 11e.2 Quantitative Neurophenomenology

Neurophenomenology (Varela et al., 1991, 2003; Lutz and Thompson 2003) proposes a bidirectional research program: phenomenological analysis of first-person experience constrains neuroscientific investigation of neural correlates, and neural findings constrain phenomenological descriptions.

The mathematical version (Lutz et al., 2002) uses **dynamical signatures** to operationalise phenomenological distinctions. In a visual task where subjects report either "prepared" or "unprepared" for a stimulus, the EEG immediately preceding the stimulus shows different dynamical signatures:

- **Prepared state**: high gamma-band synchrony (~ 40 Hz) in frontal-parietal networks, corresponding to a "ready" attentional state.
- **Unprepared state**: lower synchrony, higher alpha power (8–12 Hz), corresponding to a more internally-directed state.

The phenomenological reports and neural signatures are correlated: the same stimulus, received in different brain states, is perceived differently and processed differently.

The information-theoretic formalisation: the mutual information $\mathrm{I}(\text{phenomenal report}; \text{EEG signature})$ quantifies the degree to which the neural signal predicts the phenomenal experience. High I indicates that the neural signature is a reliable predictor of the type of experience; low I indicates that the neural measurement is not diagnostic of the phenomenal distinction.

0.94 11d. IIT and Physicalism: The Mathematical Bridge

0.94.1 11d.1 Identity Theory and Mathematical Characterisation

IIT's central claim is an **identity**: consciousness $=$ integrated information. This is a type-identity, not a token-identity: not "this particular experience is this particular physical state" but "experience in general is integrated information in general." The identity holds at the mathematical level: the mathematical structure of an experience is the same as the mathematical structure of its cause-effect power.

This is analogous to the identity of temperature with mean molecular kinetic energy. Temperature is not defined as kinetic energy, but the theoretical framework of thermodynamics reveals that temperature is kinetic energy: the concepts, developed independently, turn out to refer to the same quantity. IIT claims the same for consciousness and Φ: the framework of phenomenology (the five axioms) and the framework of causal structure (the five postulates) turn out to describe the same quantity.

The mathematical challenge is to prove that the five postulates *uniquely* determine the definition of Φ. Tononi et al. (2016) provide an argument:

1. **From the Existence Postulate**: The measure of consciousness must be positive for any system with non-trivial causal structure and zero for systems with no intrinsic causal power.
2. **From the Composition Postulate**: The measure must account for all subsets of mechanisms, not just the whole system.
3. **From the Information Postulate**: The measure must quantify how much the system's causal structure differs from the maximum entropy distribution (no constraint).

4. **From the Integration Postulate**: The measure must specifically quantify how much the causal structure is irreducible — how much it exceeds the sum of its parts.

5. **From the Exclusion Postulate**: The measure must be defined for the "dominant" cause-effect structure, not all possible structures simultaneously.

Combining these requirements, the Wasserstein distance between the intact cause-effect repertoire and the minimum-information-partition cause-effect repertoire is the essentially unique measure satisfying all five. This argument is not a rigorous mathematical proof (it relies on informal uniqueness claims), but it provides a strong motivational rationale for the specific choice of Φ.

0.94.2 11d.2 IIT's Axiomatic Derivation in Full Detail

We give the most complete presentation of the IIT derivation. Starting from axiom 4 (Integration), we derive the mathematical requirement:

Claim: The integration axiom requires that the measure ϕ of a mechanism be the minimum information distance between the intact cause-effect structure and the cause-effect structure under any partition.

Derivation. Suppose a mechanism M has cause-effect power p_{CE}. The mechanism is integrated (not decomposable into independent parts) iff for every bipartition (M_1, M_2) of M:

$$p_{\text{CE}}(M) \neq p_{\text{CE}}(M_1) \times p_{\text{CE}}(M_2).$$

The *degree* of integration is the minimum (over all bipartitions) of the distance between $p_{\text{CE}}(M)$ and $p_{\text{CE}}(M_1) \times p_{\text{CE}}(M_2)$. Using the Wasserstein-1 distance as the measure of distance (motivated by the Exclusion and Information axioms requiring a metric that respects the intrinsic geometry of the state space):

$$\phi(M) = \min_{\text{bipartition}} W_1(p_{\text{CE}}(M), p_{\text{CE}}(M_1) \times p_{\text{CE}}(M_2)).$$

This is the cause-effect version of ϕ for individual mechanisms.

The system-level Φ is then the ϕ of the whole system treated as a single mechanism, with the MIP being the partition of the *system* rather than individual mechanisms:

$$\Phi = \phi(\text{system}) = W_1(p_{\text{CE}}(\text{system}), p_{\text{CE}}(\text{MIP of system})).$$

0.94.3 11d.3 Relationship to Mutual Information and Its Failure

A natural prior proposal for the integration measure is mutual information between parts: $\phi_{\text{MI}}(M) = \text{I}(M_1; M_2)$. This has been used in several approximations to Φ (Tononi and Sporns 2003). However, IIT 3.0 explicitly rejects mutual information for three reasons:

Reason 1: Directionality. Cause-effect structures are directed (the cause of a state is different from its effect); mutual information is symmetric ($\text{I}(X;Y) = \text{I}(Y;X)$) and cannot distinguish causes from effects.

Reason 2: Extrinsic vs. intrinsic. Mutual information measures correlations relative to an *external* joint distribution; IIT requires an *intrinsic* measure that captures the system's own causal power, not correlations induced by any particular environment.

Reason 3: The sum of parts. Mutual information of a system equals the sum of mutual informations of its parts in some cases where the system clearly has complex integrated structure; the Wasserstein distance does not have this failure mode.

The specific choice of Wasserstein-1 distance (rather than KL divergence, total variation, or other distances) is motivated by the requirement that the distance should respect the intrinsic metric of the state space: moving probability mass between nearby states (nearby in Hamming distance) should cost less than moving it between distant states. KL divergence and total variation treat all states as equidistant (KL divergence does not respect the topology of the state space), while Wasserstein distance treats states as embedded in a metric space and costs proportional to the distance mass is moved.

0.95 12b. Neural Information Theory

0.95.1 12b.1 The Capacity of a Spiking Neuron

A spiking neuron transmits information about its input via a sequence of action potentials. Modelling the spike train as a binary sequence (spike or no spike in each Δt-wide time bin), the question is: how much information can a neuron transmit about its input stimulus?

For a neuron with maximum firing rate $r_{\max}$ (spikes/second), minimum rate $r_{\min} = 0$, and time window T (seconds), the maximum number of distinct spike trains is $2^{T/\Delta t}$. If we model the spike train as an independent binary string of length $N = T/\Delta t$, the maximum entropy (by the log-cardinality formula) is $\log_2(2^N) = N$ bits.

In practice, the firing rate is constrained: if the mean firing rate is $\bar{r}$ (spikes/second), the spike train is a sequence of $\bar{n} = \bar{r}T$ spikes in N bins. The number of such sequences is $\binom{N}{\bar{n}}$, and the entropy is:

$$H \approx N h_2(\bar{r}\Delta t), \qquad h_2(p) = -p\log_2 p - (1-p)\log_2(1-p),$$

where h_2 is the binary entropy function. The information rate (bits/second) is:

$$\mathcal{I} = H/T = h_2(\bar{r}\Delta t)/\Delta t \approx -\bar{r}\log_2(\bar{r}\Delta t) \text{ bits/s for small } \bar{r}\Delta t.$$

For $\bar{r} = 10$ spikes/s and $\Delta t = 1$ ms, $\mathcal{I} \approx 130$ bits/s. Typical cortical neurons transmit $\sim$ 10–100 bits/s; the theoretical maximum at physiological firing rates and typical inter-spike interval statistics is $\sim$ 300–1000 bits/s (Rieke et al., 1997).

Shannon's channel capacity theorem says: the maximum rate of reliable information transmission through a noisy channel is the channel capacity $C = \max_{p(x)} \mathrm{I}(X;Y)$, where the maximum is over all input distributions. For a Gaussian channel with noise variance σ^2 and signal power constraint P:

$$C = \frac{1}{2}\log_2\left(1 + \frac{P}{\sigma^2}\right) \text{ bits per channel use.}$$

This is the **Shannon-Hartley theorem**. In neural terms: a Gaussian neuron with signal-to-noise ratio $\mathrm{SNR} = P/\sigma^2$ can transmit $\frac{1}{2}\log_2(1 + \mathrm{SNR})$ bits per spike, and increasing SNR by adding parallel channels (more neurons coding the same signal) increases capacity at a logarithmic rate.

0.95.2 12b.2 Rate Codes versus Temporal Codes

Two hypotheses about how neurons encode information:

Rate coding: Information is encoded in the mean firing rate $\bar{r}(s) = \mathbb{E}[n|s]/T$ in response to stimulus s. The rate code is robust to noise (averaging over time reduces noise) but slow (temporal resolution $\sim T$) and wasteful (the variability of individual spikes carries no information).

Temporal coding: Information is encoded in the precise timing of individual spikes. The mutual information between the full spike train and the stimulus exceeds the mutual information between the firing rate and the stimulus: $\mathrm{I}(\text{spike train}; S) \geq \mathrm{I}(\bar{r}; S)$, with equality only if the spike timing is completely random given the rate.

Bialek et al. (1991) showed experimentally in the fly motion-sensitive neuron H1 that the spike train encodes significantly more information than the rate alone, with temporal precision down to $\sim$ 2 ms contributing additional bits. Technically, the information in the spike train is:

$$\mathrm{I}(\text{train}; S) = \int_0^\infty r(\tau) \log_2 \left(\frac{r(\tau)}{\bar{r}} \right) d\tau + \text{higher-order terms},$$

where the first term captures rate-code information and subsequent terms capture temporal correlations. Temporal coding is computationally more demanding to read out but can in principle achieve the full channel capacity of the spiking channel.

The **efficient coding hypothesis** (Barlow 1961) unifies these views: the brain should use whichever code (rate or temporal) maximises the mutual information between neural responses and stimuli, subject to metabolic and anatomical constraints. For the retina, spatial redundancy reduction via center-surround receptive fields (spatial whitening) is a form of efficient coding that discards redundant spatial correlations before sending signals to the cortex. For auditory cortex, temporal precision at the millisecond scale (phase locking) is efficient given the temporal structure of acoustic signals.

0.95.3 12b.3 Minimum Description Length: A Complete Treatment

The **Minimum Description Length** (MDL) principle of Rissanen (1978) provides a code-length interpretation of model selection that is both practically powerful and theoretically deep.

Two-part MDL. Given data D and model class $\mathcal{M} = \{M_1, M_2, \ldots\}$, the two-part MDL code assigns to each model M and its maximum-likelihood parameter $\hat{\theta}_M(D)$ the code length:

$$L_{\text{2part}}(M, D) = L(M) + L(D|\hat{\theta}_M(D)),$$

where $L(M) = -\log_2 p_0(M)$ is the code length of the model (using some universal prior p_0 over models) and $L(D|\hat{\theta}) = -\log_2 p(D|\hat{\theta})$ is the code length of the data given the model's MLE fit. Select $M^* = \arg\min_M L_{\text{2part}}(M, D)$.

Normalised Maximum Likelihood (NML). A more principled single-code MDL is the NML distribution:

$$p_{\text{NML}}(D|M) = \frac{p(D|\hat{\theta}_M(D))}{\sum_{D'} p(D'|\hat{\theta}_M(D'))},$$

which normalises the MLE distribution over all possible datasets D'. The code length under NML is:

$$L_{\text{NML}}(D|M) = -\log_2 p_{\text{NML}}(D|M) = -\log_2 p(D|\hat{\theta}) + \log_2 \mathcal{C}(M),$$

where $\mathcal{C}(M) = \sum_{D'} p(D'|\hat{\theta}_{D'}) = \int p(D'|\hat{\theta})\, dD'$ is the **parametric complexity** of model M. A model with high parametric complexity can fit many different datasets well — it is flexible but not informative.

For a k-parameter model with n i.i.d. observations, the asymptotic expansion of $\mathcal{C}(M)$ is (by the Laplace approximation):

$$\log_2 \mathcal{C}(M) \approx \frac{k}{2} \log_2 \frac{n}{2\pi e} + \log_2 \int_\Theta \sqrt{\det \mathcal{I}(\theta)}\, d\theta,$$

where $\mathcal{I}(\theta)$ is the Fisher information matrix. The first term is the Rissanen complexity: $\frac{k}{2} \log_2 n$ bits to encode a k-parameter model fitted to n data points. The second term is a model-dependent constant (the log-Section of the Fisher information geometry).

The connection to Bayesian model selection: the NML code length equals the marginal log-likelihood $-\log p(D|M)$ computed under the **Jeffreys prior** $p_J(\theta) \propto \sqrt{\det \mathcal{I}(\theta)}$ up to $O(1)$ terms. Thus MDL model selection is asymptotically equivalent to Bayesian model selection with the Jeffreys prior. The Jeffreys prior is the "least informative" prior in the information-geometric sense: it is invariant under reparametrisation of θ.

Section *VII*

Section VII — Probabilistic Reasoning and Uncertainty

0.96 14. Graphical Models: Representing Conditional Independence

0.96.1 14.1 Bayesian Networks

A **Bayesian network** (directed graphical model) is a directed acyclic graph (DAG) $\mathcal{G} = (V, E)$ together with a family of conditional probability distributions, one for each node $X_i \in V$ conditioned on its parents $\text{Pa}(X_i)$ in the graph. The joint distribution of all variables is:

$$p(x_1, \ldots, x_n) = \prod_{i=1}^{n} p(x_i \mid x_{\text{Pa}(i)}).$$

This factorisation encodes conditional independence: the **Markov condition** asserts that each variable X_i is conditionally independent of all non-descendants given its parents. The graph's topology specifies which conditional independences hold in the distribution.

The power of this representation lies in the reduction of parameters. A joint distribution over n binary variables has $2^n - 1$ free parameters. A Bayesian network with bounded in-degree d requires at most $n \cdot 2^d$ parameters — exponentially fewer. The graph is a compact encoding of the distribution's conditional independence structure.

d-separation is the graphical criterion for reading conditional independences from the DAG. Variables X and Y are d-separated given Z in $\mathcal{G}$ if every path between X and Y is **blocked** by Z. A path is blocked if it contains a non-collider in Z (a node $V \in Z$ on the path where the path passes through, not into, V), or a collider not in Z and with no descendant in Z (a node where two arrows meet, neither of which is conditioned on). The Markov condition guarantees that d-separation implies conditional independence in the corresponding distribution: $X \perp\!\!\!\perp Y \mid Z$ whenever X and Y are d-separated by Z.

0.96.2 14.2 Causal Inference: Beyond Correlation

Standard Bayesian networks represent **observational** conditional independence. **Causal inference**, developed by Pearl, Spirtes, and others, extends this to **interventional** and **counterfactual** reasoning.

An **intervention** $do(X = x)$ on variable X surgically sets $X = x$ regardless of its usual causes. In the causal graph, this corresponds to removing all edges into X and setting X to the intervened value. The **do-calculus** specifies how to compute post-intervention distributions from observational data:

$$p(Y = y \mid do(X = x)) = \sum_{z} p(Y = y \mid X = x, Z = z)\, p(Z = z),$$

where Z is any sufficient adjustment set (a set satisfying the backdoor criterion relative to X and Y). This is the **adjustment formula**: it identifies causal effects from observational data by controlling for confounders.

The do-calculus is complete: for any interventional or counterfactual query, it either provides an identification formula expressible in terms of observational distributions, or proves that identification is impossible from observational data alone. This completeness theorem establishes the exact boundary between what can and cannot be learned about causal structure without randomised experiments.

For cognition, causal inference formalises the difference between learning that A and B co-occur (correlation) and learning that A causes B (causation). Developmental evidence suggests that infants as young as

12 months distinguish these: they perform interventions (reaching and grasping) and use the results to update causal models, not merely associative ones. The formal theory provides the mathematical language for this empirical finding.

0.97 12c. Population Codes and Fisher Information

0.97.1 12c.1 Neural Tuning Curves and the Population Vector

A **population code** represents a stimulus variable θ (e.g., the direction of visual motion) through the joint activity of N neurons, each with a **tuning curve** $f_i(\theta)$ specifying the neuron's mean firing rate as a function of θ. The noisy response of neuron i is $r_i \sim \text{Poisson}(f_i(\theta))$ (or Gaussian with mean $f_i(\theta)$).

The classical **population vector decoder** (Georgopoulos et al., 1986) estimates the stimulus as:

$$\hat{\theta}_{\text{PV}} = \angle\left(\sum_{i=1}^{N} r_i \mathbf{c}_i\right),$$

where $\mathbf{c}_i = (\cos\theta_i^*, \sin\theta_i^*)$ is the **preferred direction** of neuron i (the direction that maximises f_i). This is a linear weighted sum of the neural responses, with weights given by the preferred directions.

The population vector is biased when tuning curves are asymmetric or the distribution of preferred directions is non-uniform. The maximum likelihood decoder is unbiased and efficient (achieves the Cramér-Rao bound), but requires knowing the tuning curves and the noise model.

0.97.2 12c.2 The Fisher Information Bound on Population Coding

The **Fisher information** of the population code is the key quantity bounding the precision of stimulus reconstruction. For Poisson noise with mean $f_i(\theta)$:

$$\mathcal{I}_{\text{pop}}(\theta) = \sum_{i=1}^{N} \frac{(f_i'(\theta))^2}{f_i(\theta)},$$

where $f_i'(\theta) = \partial f_i(\theta)/\partial\theta$. The Cramér-Rao bound states: for any unbiased estimator $\hat{\theta}$,

$$\text{Var}(\hat{\theta}) \geq \frac{1}{\mathcal{I}_{\text{pop}}(\theta)}.$$

The Fisher information is additive across statistically independent neurons (the sum formula), so N neurons with independent Poisson noise have Fisher information N times the single-neuron Fisher information (for identical tuning curves).

For a population with uniformly distributed preferred directions and identical Gaussian tuning curves $f_i(\theta) = A\exp(-(\theta-\theta_i^*)^2/2\sigma_{\text{tc}}^2)$, the Fisher information is approximately:

$$\mathcal{I}_{\text{pop}} \approx \frac{NA^2}{\sigma_{\text{tc}}^2 \cdot f_{\max}\sigma_{\text{tc}}} = \frac{NA}{f_{\max}\sigma_{\text{tc}}^3} \cdot A\sigma_{\text{tc}},$$

where $f_{\max}$ is the peak firing rate and σ_{tc} is the tuning width. Optimal tuning width for a fixed total energy budget ($NA \cdot \sigma_{\text{tc}} = \text{const}$) is $\sigma_{\text{tc}} \propto N^{-1/3}$ — narrower tuning curves are more informative per

neuron, but wider tuning curves are more robust to noise. This trade-off, and the predicted relationship between N, σ_{tc}, and coding precision, is a quantitative prediction of population coding theory testable by electrophysiology.

0.97.3 12c.3 Redundancy and Correlation in Neural Populations

When neurons are **correlated** (their noise covaries), the Fisher information of the population is not simply the sum of individual Fisher informations. For a multivariate Gaussian noise model with noise covariance Q and mean firing rates $\mathbf{f}(\theta)$:

$$\mathcal{I}_{\text{pop}}(\theta) = \mathbf{f}'(\theta)^\top Q^{-1} \mathbf{f}'(\theta).$$

This is the Fisher information of a multivariate Gaussian: it depends on the precision matrix Q^{-1} of the noise, not just the variances. Noise correlations can *limit* coding precision even as the population grows large (if all neurons share the same signal-correlated noise) or *be irrelevant* (if noise correlations are in the null space of the signal).

The key result (Moreno-Bote et al., 2014): noise correlations that are **aligned** with the signal (proportional to $\mathbf{f}'(\theta)\mathbf{f}'(\theta)^\top$) catastrophically limit coding precision, making the Fisher information bounded even as $N \to \infty$. Noise correlations orthogonal to the signal do not limit coding. This **information-limiting correlation** hypothesis predicts that cortical noise correlations are partially aligned with the signal, explaining why large neural populations in sensory cortex do not provide arbitrarily precise estimates.

For the binding problem (Section 13b.1), correlated noise across feature-coding neurons may actually be beneficial: if the neurons coding for the colour and shape of the same object have correlated activity, their joint response provides more information about the object than their independent responses, implementing a form of binding through correlation structure.

0.98 12d. Kolmogorov Complexity: Complete Theory and Cognitive Applications

0.98.1 12d.1 Kolmogorov Complexity: Definitions and Invariance

The **Kolmogorov complexity** $K(x)$ of a binary string $x \in \{0,1\}^*$ (with respect to a universal Turing machine U) is the length of the shortest program p such that $U(p) = x$:

$$K(x) = \min_{p : U(p) = x} |p|.$$

$K(x)$ measures the **minimum description length** of x: the shortest algorithm that produces x.

Theorem 0.98.1 (Invariance Theorem)**.** For any two universal Turing machines U and V, there exists a constant c such that for all x:

$$|K_U(x) - K_V(x)| \leq c.$$

Proof. U can simulate V by prepending a finite program for V's interpreter to any V-program; this increases length by at most $|V\text{-interpreter}|$ bits. By symmetry, $K_V(x) \leq K_U(x) + c_{VU}$ as well. □

The invariance theorem establishes that Kolmogorov complexity is a machine-independent concept (up to additive constants): the choice of universal machine affects $K(x)$ by at most a constant. This is the

algorithmic analogue of the choice of reference in information theory (the choice of log base).

Key properties:

- $K(x) \leq |x| + c$: any string is at most self-describing.
- Most strings have $K(x) \approx |x|$: random (incompressible) strings.
- $K(x, y) \leq K(x) + K(y) + c$: joint complexity is at most the sum of marginals.
- $K(x|y) = K(x, y) - K(y) + O(\log K(x, y))$: conditional complexity.

Uncomputability. $K(x)$ is not computable: there is no algorithm that, given x, returns $K(x)$. This follows from the halting problem: if K were computable, we could enumerate all programs in order of length and decide the halting problem. However, K is **upper semi-computable**: there is an algorithm that converges to $K(x)$ from above (by enumerating shorter and shorter programs that produce x).

0.98.2 12d.2 Algorithmic Mutual Information and the Minimum Description Length Theorem

The **algorithmic mutual information** between x and y is:

$$I(x : y) = K(x) - K(x|y^*) = K(y) - K(y|x^*) + O(\log K(x) + \log K(y)),$$

where y^* is the shortest program for y. Unlike Shannon mutual information (which is defined for distributions), algorithmic mutual information is defined for individual strings and measures the *shared algorithmic structure* between x and y.

The **symmetry of information** theorem: $K(x|y^*) + K(y) = K(x, y) + O(\log K(x, y))$, so $I(x : y) \approx I(y : x)$ up to logarithmic terms. This is the algorithmic analogue of the symmetry $\mathrm{I}(X; Y) = \mathrm{I}(Y; X)$ of Shannon mutual information.

MDL and model selection. For a model class $\mathcal{M}$ and data D, the two-part MDL code length is:

$$L(M, D) = K(M) + K(D|M),$$

where $K(M)$ is the complexity of the model and $K(D|M)$ is the complexity of the data given the model. Minimising $L(M, D)$ over all models selects the model that provides the shortest *total description* of the data (model plus residual).

The **MDL theorem** (Li and Vitányi 1997): for any computable probability distribution P over data sequences, the MDL predictor (using the model with shortest total description) is optimal in the following sense: its cumulative regret relative to the best computable predictor is $O(K(P))$. This is an analogue of Solomonoff's prediction theorem for model selection rather than pure prediction.

0.98.3 12d.3 Minimum Message Length and Human Category Learning

Minimum Message Length (MML; Wallace and Boulton 1968) is the Bayesian MDL variant that accounts for the prior in the code length. The MML code assigns to model M and data D:

$$I(M, D) = -\log p(M) - \log p(D|M, \hat{\theta}_M) + \frac{d}{2} \log \frac{n}{2\pi e} + \text{const},$$

where d is the number of model parameters. MML penalises models not only by their number of parameters but also by their prior probability, making it equivalent to MAP estimation with a correction for model complexity.

For **human category learning**, MML provides a normative account of why humans form categories of intermediate specificity (neither too broad nor too narrow): categorisation at the optimal MDL level minimises the total code length for labelling instances. Empirically, the optimal MDL level predicts the **basic level effect** in categorisation: "dog" is more basic than "mammal" (too general) or "Labrador" (too specific) because the description length for a "dog" image is shortest at the basic level, where perceptual features are maximally diagnostic of category membership.

0.99 14b. Belief Propagation and Junction Tree Algorithm

0.99.1 14b.1 The Sum-Product Algorithm

Inference in a Bayesian network—computing marginal distributions $p(x_i)$ or conditional distributions $p(x_i|\text{evidence})$—is generally #P-hard. For tree-structured graphical models, however, exact inference can be performed in linear time using the **belief propagation** (sum-product) algorithm.

On a tree factor graph with variable nodes x_i and factor nodes f_a, define **messages** passed between adjacent nodes. The message from variable x_i to factor f_a at iteration t is:

$$\mu_{x_i \to f_a}^{(t)}(x_i) = \prod_{b \in \mathcal{N}(x_i) \setminus a} \mu_{f_b \to x_i}^{(t-1)}(x_i),$$

and from factor f_a to variable x_i:

$$\mu_{f_a \to x_i}^{(t)}(x_i) = \sum_{\mathbf{x}_{a \setminus i}} f_a(\mathbf{x}_a) \prod_{j \in \mathcal{N}(f_a) \setminus i} \mu_{x_j \to f_a}^{(t)}(x_j),$$

where $\mathcal{N}(\cdot)$ denotes neighbours in the factor graph and $\mathbf{x}_{a \setminus i}$ ranges over all variables in the scope of f_a except x_i.

Theorem 0.99.1 (Belief propagation exactness on trees)**.** On a tree factor graph, the belief propagation algorithm with two passes (leaves to root, root to leaves) converges in $O(\text{diameter})$ iterations and computes exact marginals:

$$b(x_i) \propto \prod_{a \in \mathcal{N}(x_i)} \mu_{f_a \to x_i}(x_i) = p(x_i) \text{ for all } i.$$

Proof sketch. The messages passed from leaves inward are exactly the conditional distributions of the subtree below each node, given the parent. At any node, the product of all incoming messages is proportional to the marginal distribution at that node: the marginalisation is exact because the tree structure prevents double-counting of paths (no cycles). A formal proof by induction on the tree depth establishes this for all nodes. □

For cyclic graphs (which include almost all interesting models), belief propagation is applied *as if* the graph were a tree: the algorithm still terminates (on loopy graphs, convergence is not guaranteed but empirically often occurs) and its fixed points, when they exist, can be characterised as the stationary

points of the **Bethe free energy**:

$$\mathcal{F}_{\text{Bethe}} = -\sum_{a}\sum_{\mathbf{x}_a} b_a(\mathbf{x}_a)\ln\frac{f_a(\mathbf{x}_a)}{b_a(\mathbf{x}_a)} - \sum_{i}(1-d_i)\sum_{x_i} b_i(x_i)\ln b_i(x_i),$$

where b_a are factor beliefs, b_i are variable beliefs, and $d_i = |\mathcal{N}(x_i)|$ is the degree of variable i. The Bethe free energy is an approximation to the variational free energy that decomposes the model into overlapping neighbourhood subgraphs.

0.99.2 14b.2 The Junction Tree Algorithm

The **junction tree algorithm** (also: clique tree algorithm, or variable elimination organised as message passing) provides exact inference in general (non-tree) graphical models, at the cost of an exponential dependence on **treewidth**—a measure of the graph's complexity.

Step 1: Moralise and triangulate. Given a Bayesian network $\mathcal{G}$, form the moral graph by connecting the parents of each node and dropping edge directions, then triangulate (add fill-in edges until every cycle of length ≥ 4 has a chord).

Step 2: Extract cliques. The maximal cliques of the triangulated moral graph form the nodes of the junction tree. The clique tree is connected by **separator sets** $S_{ij} = C_i \cap C_j$ for adjacent cliques C_i and C_j.

Step 3: Run belief propagation on the clique tree. The clique tree is a tree of cliques, and belief propagation on it is exact (by the preceding theorem). Messages are now distributions over *sets of variables* rather than individual variables:

$$\mu_{C_i \to C_j}(S_{ij}) = \sum_{C_i \setminus S_{ij}} \psi_{C_i}(\mathbf{x}_{C_i}) \prod_{k \in \mathcal{N}(C_i)\setminus j} \mu_{C_k \to C_i}(S_{ki}),$$

where $\psi_{C_i}(\mathbf{x}_{C_i}) = \prod_{a:\text{scope}(f_a)\subseteq C_i} f_a(\mathbf{x}_a)$ is the product of all factors whose scope is contained in clique C_i.

The computational cost is $O(nk^{w+1})$ where w is the **treewidth** (one less than the maximum clique size after triangulation) and k is the number of states per variable. For graphs with low treewidth (chains, trees, grids, planar graphs), this is tractable. For graphs with high treewidth (complete graphs, random graphs with high edge density), treewidth equals $n-1$ and the algorithm reduces to brute-force marginalisation.

The treewidth of graphs is NP-hard to compute exactly but efficiently approximated by **greedy triangulation** heuristics (minimum fill-in, minimum degree). The importance of treewidth for cognitive inference is the structural claim: the neural circuits used for exact inference in the brain should correspond to graphical models with low treewidth, favouring hierarchical and locally connected architectures.

0.100 14f. Causal Inference in Cognitive Neuroscience

0.100.1 14f.1 The Problem of Reverse Inference

Reverse inference in neuroimaging is the logical fallacy of inferring a mental state from a brain activation: "Area X was active; area X is associated with process P; therefore, process P was engaged." This argument

is invalid because areas are not selective: the same area may be active for multiple processes.

Formally, reverse inference is the inference from $P(\text{activation X}|\text{process P})$ (the forward inference: what activation does process P cause?) to $P(\text{process P}|\text{activation X})$ (the reverse: what process caused this activation?). By Bayes' theorem:

$$P(\text{P}|\text{X}) = \frac{P(\text{X}|\text{P}) \cdot P(\text{P})}{P(\text{X})}.$$

The validity of the reverse inference depends on the **selectivity** of area X for process P: if $P(\text{X}|\text{not-P})$ is high (many other processes also activate X), the reverse inference is weak even if $P(\text{X}|\text{P})$ is near 1.

Poldrack's meta-analytic approach (2006) uses a large database of neuroimaging studies to estimate $P(\text{X}|\text{P})$ and $P(\text{P})$ empirically, enabling quantitative reverse inference. For the insula, $P(\text{insula active}|\text{pain task}) = 0.82$ but $P(\text{insula active}|\text{emotion task}) = 0.79$ — so insula activation is not selective for pain. Bayes' theorem gives $P(\text{pain}|\text{insula}) \approx P(\text{pain})$ (near base rate).

0.100.2 14f.2 Granger Causality and its Limitations

Granger causality (Granger 1969) defines a statistical sense of causality: time series X Granger-causes Y if past values of X improve the prediction of Y beyond what is possible with past values of Y alone. Formally, X GC-causes Y iff:

$$\text{Var}(\epsilon_t^Y | Y_{t-1}, Y_{t-2}, \ldots) > \text{Var}(\epsilon_t^Y | Y_{t-1}, Y_{t-2}, \ldots, X_{t-1}, X_{t-2}, \ldots),$$

where ϵ_t^Y is the prediction error for Y at time t.

For multivariate time series, GC can be computed via vector autoregressive (VAR) models. A VAR(p) model:

$$\mathbf{y}_t = \sum_{k=1}^{p} A_k \mathbf{y}_{t-k} + \boldsymbol{\epsilon}_t,$$

estimates the coefficient matrices A_k by OLS. Granger causality from variable j to variable i is then assessed by testing whether the j-th column of any A_k is significantly non-zero (F-test or likelihood ratio test).

Limitations in neuroscience. GC is not causal in the Pearl/interventionist sense. It detects *predictive precedence*, not mechanistic causation:

1. **Common driver**: If regions A and B are both driven by a common input C, A may GC-cause B (or vice versa) even if neither directly affects the other.

2. **Downsampling**: fMRI BOLD signals are sampled at $\sim$1–2 Hz, while neural dynamics operate at milliseconds. GC on downsampled data gives misleading results about the true causal direction.

3. **Hemodynamic variability**: The BOLD hemodynamic response function (HRF) varies across brain regions; different HRF delays introduce spurious GC even between simultaneously active regions.

The do-calculus (Section 14c) and structural causal models provide a principled alternative: instead of testing predictive precedence, model the causal structure explicitly and use interventional data (TMS, lesions, optogenetics) to identify causal effects.

0.100.3 14f.3 TMS as a Causal Intervention

Transcranial Magnetic Stimulation (TMS) provides a near-ideal implementation of the do-calculus's intervention operator $do(X = x)$: applying TMS to cortical area X forces its activity to a desired level ($\sim$suppression at $\sim$10 Hz rTMS), allowing the estimation of $P(Y|do(X = x))$ rather than the observational $P(Y|X = x)$.

For testing a causal hypothesis about cortical hierarchy (e.g., "V1 activity is causally upstream of V5 in motion perception"), the experimental design is:

1. Condition 1 (observation): Present motion stimulus, measure V1 and V5 activity via fMRI/EEG.
2. Condition 2 (intervention): Apply TMS to V1 during motion stimulus, measure V5 activity.

If V5 activity decreases in Condition 2 (TMS to V1) relative to Condition 1 (no TMS), this is evidence that V1 causally drives V5: $P(V5|do(V1 = \text{low})) < P(V5|\text{no intervention})$.

The **intervention contrast** $\text{IC}(V1 \to V5) = P(V5|do(V1 = \text{high})) - P(V5|do(V1 = \text{low}))$ estimates the causal effect size. This is identifiable from the TMS experiment without knowledge of the full causal graph (unlike Pearl's do-calculus, which requires the full graph structure for identification from observational data only).

0.101 14d. Variable Elimination and the Complexity of Inference

0.101.1 14d.1 Variable Elimination

Variable elimination is the fundamental exact inference algorithm for graphical models. It computes a marginal $p(x_Q)$ for a query variable X_Q by summing out all other variables in a specific order:

$$p(x_Q) = \sum_{x_1} \sum_{x_2} \cdots \sum_{x_{n-1}} \prod_{i=1}^{n} \psi_i(x_{\text{scope}(i)}),$$

where ψ_i are factors (CPDs in a Bayesian network, or potential functions in an MRF) and the summations are applied in an order that minimises intermediate factor size.

The key operation is **sum-product**:

1. Initialise the factor list $\Phi = \{\psi_1, \ldots, \psi_m\}$.
2. For each variable X_j not in the query (in the elimination order):
   (a) Collect all factors $\Phi_j \subseteq \Phi$ whose scope includes X_j.
   (b) Multiply them: $\tau_j(x_{\text{scope}(\Phi_j)\setminus j}) = \sum_{x_j} \prod_{\psi \in \Phi_j} \psi$.
   (c) Replace Φ_j with τ_j in Φ.
3. The remaining factor is (proportional to) $p(x_Q)$.

The cost of each multiplication-summation step is exponential in the number of variables in the combined scope of Φ_j. The **induced width** (treewidth) of a graph is the maximum number of variables in any

single elimination step, over the optimal ordering. Variable elimination with the optimal ordering costs $O(nk^{w+1})$ where k is the domain size and w is the treewidth.

Finding the optimal elimination ordering is NP-hard (it is equivalent to the minimum fill-in problem, which in turn is equivalent to minimum treewidth). Heuristics like **minimum degree** (eliminate the variable connected to fewest remaining variables) and **minimum fill-in** (eliminate the variable requiring fewest new edges) give good approximations in practice.

0.101.2 14d.2 Approximate Inference: Loopy Belief Propagation

For graphs with high treewidth, exact inference is intractable. **Loopy belief propagation** (LBP) applies the sum-product algorithm to loopy graphs by simply ignoring the loops:

At each iteration t, update all messages simultaneously (or in sequence):

$$\mu_{i\to j}^{(t+1)}(x_j) \propto \sum_{x_i} \psi_{ij}(x_i, x_j)\psi_i(x_i) \prod_{k\in\mathcal{N}(i)\setminus j} \mu_{k\to i}^{(t)}(x_i).$$

The approximate beliefs are:

$$b_i(x_i) \propto \psi_i(x_i) \prod_{j\in\mathcal{N}(i)} \mu_{j\to i}(x_i), \qquad b_{ij}(x_i, x_j) \propto \psi_{ij}(x_i, x_j)\psi_i(x_i)\psi_j(x_j) \prod_{k\neq j} \mu_{k\to i} \prod_{l\neq i} \mu_{l\to j}.$$

LBP has no convergence guarantee on loopy graphs, but when it converges, its fixed points are the stationary points of the Bethe free energy:

$$\mathcal{F}_B = -\sum_{(i,j)\in E} \sum_{x_i,x_j} b_{ij} \log \frac{\psi_{ij}}{b_{ij}} - \sum_i \sum_{x_i} b_i \log \frac{\psi_i}{b_i^{d_i}},$$

where $d_i = |\mathcal{N}(i)|$ is the degree of node i and $b_i^{d_i} = \prod_{j\in\mathcal{N}(i)} b_i$ (the belief raised to the degree power). The Bethe free energy is the mean-field free energy for a tree-structured approximation of the loopy graph.

Empirically, LBP works extremely well on sparse graphs (LDPC codes in communications, stereo vision in computer vision) and moderately well on dense graphs. Its failures are most pronounced when loops are short (the same information circulates repeatedly, amplifying errors) and when likelihoods are sharp (high-confidence observations make approximations more sensitive to errors).

0.102 14h. Variational Inference for Causal Models

0.102.1 14h.1 Structural Causal Models and Posterior Inference

A **Structural Causal Model** (SCM; Pearl 2000) specifies a set of structural equations $X_i = f_i(PA_i, U_i)$ where PA_i are the parents of X_i in a DAG $\mathcal{G}$, U_i are exogenous (noise) variables, and f_i are deterministic functions. When the U_i are mutually independent, the SCM defines a unique joint distribution over the observable variables.

Given an SCM and data $D = (x^{(1)}, \ldots, x^{(n)})$, the Bayesian approach places priors $p(\mathbf{f}, \sigma_U^2)$ over the

structural functions $\mathbf{f} = (f_1, \ldots, f_d)$ and noise variances, and computes the posterior:

$$p(\mathbf{f}, \sigma_U^2 | D) \propto p(D|\mathbf{f}, \sigma_U^2) p(\mathbf{f}, \sigma_U^2).$$

For linear Gaussian SCMs ($f_i = \beta_i^\top PA_i + U_i$, $U_i \sim \mathcal{N}(0, \sigma_i^2)$), the posterior over the coefficient matrices β is Gaussian (conjugate). For nonlinear SCMs, variational inference (Section 5) or MCMC is required.

GP-SCM. A natural nonparametric prior for structural functions is the GP: $f_i | k \sim \mathcal{GP}(0, k)$ independently for each i. The GP-SCM places a flexible nonparametric prior over the causal mechanisms. Posterior inference over the functions f_i and the causal structure $\mathcal{G}$ simultaneously is performed by MCMC over graphs (structure MCMC) combined with GP posterior updates for each proposed graph.

0.102.2 14h.2 Amortised Causal Inference

For large-scale causal discovery (many variables, many observations), amortised approaches replace MCMC with trained neural networks that directly output the posterior over causal structures. **DiBS** (Differentiable Bayesian Structure discovery; Lorch et al., 2021) parametrises the distribution over DAGs using continuous relaxations of the adjacency matrix:

$$P(G|\mathbf{Z}) = \prod_{i<j} \text{Bernoulli}(\sigma(\mathbf{z}_i \cdot \mathbf{z}_j)),$$

where $\mathbf{Z} = (\mathbf{z}_1, \ldots, \mathbf{z}_d)$ are latent vectors for each node, and $\sigma(\mathbf{z}_i \cdot \mathbf{z}_j)$ is the probability of an edge from j to i. Training DiBS maximises the ELBO of the joint model $p(D, G)$ over the variational distribution $q(G|\mathbf{Z})$.

The acyclicity constraint (DAGs have no directed cycles) is enforced by a differentiable penalty $h(A) = \text{tr}(e^{A \odot A}) - d$ (Zheng et al., 2018): $h(A) = 0$ iff A is acyclic. This allows gradient-based optimisation of the continuous relaxation of the DAG space, enabling scalable causal structure learning.

0.103 14g. Graph Neural Networks and Relational Reasoning

0.103.1 14g.1 Graph Neural Networks as Probabilistic Inference

Graph Neural Networks (GNNs; Scarselli et al., 2009; Kipf and Welling 2017) process data with relational structure (molecules, social networks, scene graphs) by iteratively updating node representations based on neighbour information. GNNs have a deep connection to belief propagation on graphical models.

The general GNN layer:

$$h_i^{(l+1)} = \phi\left(h_i^{(l)}, \bigoplus_{j \in \mathcal{N}(i)} \psi(h_i^{(l)}, h_j^{(l)}, e_{ij}) \right),$$

where $h_i^{(l)}$ is the representation of node i at layer l, $\mathcal{N}(i)$ are its neighbours, e_{ij} is the edge feature, ψ is the **message function**, $\bigoplus$ is the **aggregation** (sum, max, or mean), and ϕ is the **update function**.

Connection to belief propagation. The BP message from factor f_a to variable x_i (Section 14b) is:

$$\mu_{f_a \to x_i}(x_i) = \sum_{x_{\mathcal{N}(f_a)\setminus i}} f_a(x_{\mathcal{N}(f_a)}) \prod_{j \in \mathcal{N}(f_a)\setminus i} \mu_{x_j \to f_a}(x_j).$$

This is a GNN layer where: node representations are messages $\mu_{x_j \to f_a}$, aggregation is the product of incoming messages, and the message function involves the factor potential f_a. A GNN with L layers implements L iterations of BP, inheriting BP's ability to propagate information across the graph.

Graph attention networks (GAT). Replace the fixed aggregation with attention-weighted aggregation:

$$h_i^{(l+1)} = \sigma\left(\sum_{j \in \mathcal{N}(i)\cup\{i\}} \alpha_{ij} W h_j^{(l)} \right), \quad \alpha_{ij} = \frac{\exp(\text{LeakyReLU}(\mathbf{a}^\top [Wh_i \| Wh_j]))}{\sum_{k \in \mathcal{N}(i)} \exp(\dots)},$$

where α_{ij} is the attention coefficient from node j to node i, $\mathbf{a}$ is a learnable attention vector, and W is a weight matrix. The attention mechanism selects the most relevant neighbours, analogous to the precision-weighted message passing of Active Inference.

0.103.2 14g.2 Relational Reasoning in Cognitive Science

Relational reasoning — the ability to represent and manipulate structured relationships between objects — is considered a hallmark of human cognition (Holyoak and Thagard 1996; Gentner 1983). GNNs provide a computational framework for relational reasoning that bridges connectionist and symbolic approaches.

Analogy as graph isomorphism. Analogical reasoning ("hand:glove :: foot:?") requires finding a structural mapping between two relational structures. This is equivalent to finding a (partial) graph isomorphism between the source and target relational graphs. GNNs with node-level representations compute structural encodings that can be matched: two nodes are analogous if their GNN representations are similar, capturing the idea that analogous nodes occupy similar structural positions in their respective graphs.

The Neural Theorem Prover (NTP; Rocktäschel and Riedel 2017) combines GNNs with differentiable backward chaining to perform symbolic reasoning over knowledge graphs. The backward chaining rules (if $P(x, y)$ and $Q(y, z)$ then $R(x, z)$) are instantiated by soft matching of entity embeddings, and the GNN propagates inference through the knowledge graph. NTP achieves higher accuracy than pure symbolic reasoners on incomplete knowledge graphs, because the distributed representations allow generalisation to unseen entities.

0.104 14e. D-Separation, Markov Blankets, and Conditional Independence

0.104.1 14e.1 D-Separation: Complete Algorithm

D-separation (Pearl 1988) is the criterion for reading off conditional independence relations from a directed acyclic graph (DAG). Three nodes X, Y, Z are **d-separated** by a set W (written $X \perp\!\!\!\perp_G Y | W$) iff every **path** (undirected) between X and Y is **blocked** by W.

A path $X = V_0, V_1, \dots, V_k = Y$ is **blocked** by W iff there exists a node V_i on the path such that:

- V_i is a **non-collider** on the path ($\leftarrow V_i \rightarrow$ or $\leftarrow V_i \leftarrow$ or $\rightarrow V_i \rightarrow$) and $V_i \in W$ (the non-collider is observed).
- V_i is a **collider** on the path ($\rightarrow V_i \leftarrow$) and $V_i \notin W$ and no descendant of V_i is in W (the collider is not observed and no descendant is observed).

Theorem 0.104.1 (Soundness and Completeness; Verma and Pearl 1988)**.** In any DAG G and for any probability distribution P that is **Markov** w.r.t. G (i.e., satisfies all d-separation constraints as conditional independences), d-separation is:

- **Sound**: if $X \perp\!\!\!\perp_G Y|W$, then $X \perp\!\!\!\perp_P Y|W$.
- **Complete**: if $X \not\perp\!\!\!\perp_G Y|W$ (they are d-connected given W), then there exists a distribution P that is Markov w.r.t. G such that $X \not\perp\!\!\!\perp_P Y|W$.

The completeness result says: d-separation captures all and only the conditional independence relations that hold in every distribution compatible with the graph. Any additional independence (not implied by d-separation) would be a "special" independence requiring extra parametric constraints.

Algorithm. The Bayes Ball algorithm (Shachter 1998) determines all variables d-separated from a set X given evidence W in $O(|V| + |E|)$ time (linear in graph size). It simulates the propagation of "balls" through the graph, with non-colliders passing balls in one direction and colliders passing balls only when observed.

0.104.2 14e.2 Faithfulness and Causal Discovery

A distribution P is **faithful** to DAG G if the conditional independence relations in P are exactly those implied by d-separation in G (no extra independences). Under faithfulness, the DAG structure is identifiable from the distribution P up to the **Markov equivalence class**: the set of DAGs with the same skeleton (undirected edges) and the same **v-structures** ($\rightarrow Z \leftarrow$ patterns).

Markov equivalence is characterised by the **CPDAG** (completed partially directed acyclic graph): a mixed graph with both directed and undirected edges, where directed edges indicate the common orientation across all DAGs in the equivalence class, and undirected edges indicate orientations that vary within the class.

The **PC algorithm** (Peter and Clark, implemented by Spirtes et al., 1993) recovers the CPDAG from observational data:

1. **Skeleton learning**: Test all pairs of variables for marginal independence; remove edges between independent pairs.
2. **Conditioning**: For remaining edges, test for conditional independence given all subsets of adjacent variables; remove edges when a separating set is found.
3. **Orientation**: Orient v-structures ($X - Z - Y$ with X and Y non-adjacent and $Z \notin \text{sep}(X, Y)$) as $X \rightarrow Z \leftarrow Y$; propagate orientations using Meek's rules.

PC is consistent (recovers the true CPDAG in the large-sample limit) under faithfulness. FCI (Fast Causal Inference) extends PC to handle hidden variables, producing a PAG (partial ancestral graph) that represents the equivalence class of acyclic directed mixed graphs compatible with the observed independences.

0.105 14c. The Do-Calculus: Complete Rules and Applications

0.105.1 14c.1 The Three Rules of the Do-Calculus

Pearl's do-calculus (Pearl 1995, 2000) provides a complete axiomatisation of interventional and counterfactual reasoning. For a causal model with graph $\mathcal{G}$ and an intervention graph $\mathcal{G}_{\overline{X}}$ obtained by deleting all edges into X (representing the intervention $do(X = x)$):

Theorem 0.105.1 (Do-Calculus; Pearl 1995)**.** Let X, Y, Z, W be disjoint sets of variables. The following rules hold for any probability distribution compatible with $\mathcal{G}$:

Rule 1 (Insertion/deletion of observations):

$$p(y \mid do(x), z, w) = p(y \mid do(x), w) \qquad \text{if } (Y \perp\!\!\!\perp Z \mid X, W)_{\mathcal{G}_{\overline{X}}},$$

i.e., if Z is d-separated from Y given (X, W) in the graph $\mathcal{G}_{\overline{X}}$ obtained by removing edges into X.

Rule 2 (Action/observation exchange):

$$p(y \mid do(x), do(z), w) = p(y \mid do(x), z, w) \qquad \text{if } (Y \perp\!\!\!\perp Z \mid X, W)_{\mathcal{G}_{\overline{X},\underline{Z}}},$$

where $\mathcal{G}_{\overline{X},\underline{Z}}$ is obtained by removing edges into X and removing edges out of Z.

Rule 3 (Insertion/deletion of actions):

$$p(y \mid do(x), do(z), w) = p(y \mid do(x), w) \qquad \text{if } (Y \perp\!\!\!\perp Z \mid X, W)_{\mathcal{G}_{\overline{X},\overline{Z(W)}}},$$

where $Z(W)$ is the subset of Z not in any ancestor of W in $\mathcal{G}_{\overline{X}}$, and $\mathcal{G}_{\overline{X},\overline{Z(W)}}$ removes edges into both X and $Z(W)$.

Theorem 0.105.2 (Completeness of the Do-Calculus; Shpitser and Pearl 2006)**.** A causal effect $p(y \mid do(x))$ is identifiable from observational data in a causal graph $\mathcal{G}$ if and only if the do-calculus can reduce it to an expression in observational probabilities alone.

This completeness result establishes that the do-calculus is not merely a sufficient tool for identification: it is a complete characterisation of what is and is not identifiable from observational data, given the causal structure.

0.105.2 14c.2 The Identification Algorithm

The ID algorithm (Shpitser and Pearl 2006) systematically applies the do-calculus to either identify $p(y|do(x))$ or determine that it is non-identifiable. The algorithm proceeds by:

1. **Simplify**: Remove variables irrelevant to the query (not ancestors of Y given X).
2. **C-component decomposition**: Factor the interventional distribution over the C-components (connected components of the bidirected edges) of $\mathcal{G}$.
3. **Recursive identification**: For each C-component, apply the truncated factorisation formula $p(v|do(x)) = \prod_{i:V_i \notin X} p(v_i|pa_i)$ (where pa_i are the parents of V_i in $\mathcal{G}$, with the intervention variables fixed) and marginalise.

4. **Non-identification**: If any C-component contains a bidirected path (hidden common cause) that cannot be "blocked" by the observed variables, declare non-identifiability and output "FAIL".

The key insight is that a bidirected edge $U \leftrightarrow V$ (representing an unobserved common cause) blocks identification of $p(y|do(x))$ when U and V lie on a path from X to Y that cannot be blocked by observed variables. When such paths exist, no observational data — no matter how large — can identify the causal effect.

0.105.3 14c.3 The Ladder of Causation

Pearl's **ladder of causation** organises causal queries into three levels of increasing expressiveness:

Level 1: Association ($P(y|x)$). Observational questions: what is the probability of Y given that we observe $X = x$? This is standard conditional probability. A regression, a neural network, or a Bayesian classifier operates at Level 1.

Level 2: Intervention ($P(y|do(x))$). Interventional questions: what would happen to Y if we set $X = x$ by external force? This requires a causal model. Randomised controlled trials provide Level 2 data. The do-calculus computes Level 2 quantities from Level 1 data when the causal structure is known.

Level 3: Counterfactuals ($P(Y_x = y|X = x', Y = y')$). Retrospective questions: given that we observed $X = x'$ and $Y = y'$, what would Y have been had X been set to x instead? Counterfactuals require the full structural causal model (individual-level equations, not just population-level distributions).

No algorithm operating only at a lower level can answer questions at a higher level. This is the **interventional barrier** between Levels 1 and 2 (correlation does not imply causation), and the **counterfactual barrier** between Levels 2 and 3. Cognitive agents capable of counterfactual reasoning operate at Level 3 of the ladder.

0.105.4 14c.4 Cognitive Development and Causal Learning

The acquisition of causal knowledge in children has been extensively studied using interventionist methods (Gopnik et al., 2004). Children as young as 2–3 years distinguish interventions from observations (Rule 2 of the do-calculus) and use causal models to predict the effects of novel interventions (Schulz et al., 2007).

Mathematically, Bayesian structure learning over causal graphs models this process: the learner maintains a posterior distribution $p(\mathcal{G}|D)$ over causal graphs, updating it as observational and interventional data arrive. The likelihood of data D given graph $\mathcal{G}$ is $p(D|\mathcal{G}) = \prod_t p(x_t^{\text{obs}}|\mathcal{G}) \prod_s p(x_s^{\text{int}}|do(x_s^{\text{int,do}}), \mathcal{G})$, combining the observational and interventional likelihoods via the do-calculus. Learning the causal graph from scratch requires a prior over graphs that penalises complexity (sparse graphs) and favours acyclicity.

Section *VIII*

Section VIII — Non-Classical Logics and Defeasible Reasoning

0.106 15. The Defeasibility of Rational Inference

Classical logic is monotone: if $\Gamma \vdash \varphi$ (the conclusion φ follows from premises Γ), then $\Gamma, \psi \vdash \varphi$ for any additional premise ψ. Adding more information cannot retract conclusions already derived.

Human reasoning is not monotone. We routinely draw conclusions that are later retracted when more information arrives. "Tweety is a bird; birds fly; therefore Tweety flies" is overturned when we learn Tweety is a penguin. "This patient has fever; fever suggests infection; therefore prescribe antibiotics" is retracted when we learn the fever is drug-induced. The willingness to draw tentative conclusions from incomplete information, and to revise them gracefully in the light of counterevidence, is a hallmark of practical reasoning.

Formalising this requires logics that are non-monotone — where adding premises can retract conclusions. This Section surveys the major approaches.

0.107 16. Default Logic and Non-Monotonic Reasoning

0.107.1 16.1 Default Rules

Reiter's **default logic** extends classical logic with **default rules** of the form:

$$\frac{\alpha : \beta_1, \ldots, \beta_m}{\gamma}$$

Read: "if α is derivable, and $\beta_1, \ldots, \beta_m$ are all consistent with what is known, then derive γ." The formula α is the **prerequisite**, the β_i are the **justifications**, and γ is the **consequent**.

The canonical example: $\frac{\text{Bird}(x) : \text{Flies}(x)}{\text{Flies}(x)}$. If x is a bird, and it is consistent to assume x flies, then conclude x flies. This is a defeasible inference: if we later learn that x is a penguin and penguins do not fly, the justification $\text{Flies}(x)$ becomes inconsistent with our knowledge, and the default rule fires no longer.

The semantics of default logic are defined via **extensions**: a set E of sentences is an extension of a default theory $(\mathcal{W}, \mathcal{D})$ (where $\mathcal{W}$ is the set of hard facts and $\mathcal{D}$ is the set of default rules) if E is the smallest set satisfying: $\mathcal{W} \subseteq E$; E is deductively closed (closed under classical entailment); for every default $\alpha : \beta_1, \ldots, \beta_m / \gamma \in \mathcal{D}$, if $\alpha \in E$ and $\neg\beta_i \notin E$ for all i, then $\gamma \in E$.

Extensions may not be unique: a default theory may have multiple extensions, corresponding to different "possible worlds" consistent with the defaults. A sentence is **sceptically** entailed if it is in every extension, and **credulously** entailed if it is in some extension. Sceptical reasoning is conservative; credulous reasoning is more liberal.

0.107.2 16.2 Modal Epistemic Logic

Modal logic extends classical propositional logic with operators $\Box$ ("necessarily" or "it is known that") and $\Diamond$ ("possibly" or "it is consistent that"). The **normal modal logics** are axiom systems extending the K axiom:

$$\Box(\varphi \Rightarrow \psi) \Rightarrow (\Box\varphi \Rightarrow \Box\psi).$$

Different additional axioms characterise different notions of knowledge or necessity. The axiom T : $\Box\varphi \Rightarrow \varphi$ (what is known is true) gives the logic **KT**. Adding $4 : \Box\varphi \Rightarrow \Box\Box\varphi$ (if you know it, you know

you know it) gives **S4**. Adding $5 : \Diamond\varphi \Rightarrow \Box\Diamond\varphi$ gives **S5**, the logic of **absolute knowledge** in which what is known is independent of the agent's introspective limitations.

The **possible worlds semantics** (Kripke semantics) assigns to each formula a set of worlds at which it is true. $\Box\varphi$ is true at world w iff φ is true at all worlds w' accessible from w via the accessibility relation R. Different modal logics correspond to different properties of R: reflexivity gives T; transitivity gives 4; symmetry gives B; reflexivity, transitivity, and symmetry (equivalence relation) gives $S5$.

For cognitive science, epistemic modal logic is the formal language for reasoning about knowledge and belief in multi-agent settings: what does agent A know, what does A know about B's knowledge, and what do both agents know that they both know (common knowledge)? The logic of common knowledge, captured by the fixed-point axiom $C\varphi \Leftrightarrow E(C\varphi \wedge \varphi)$ (where $E\varphi$ means every agent knows φ), is the formal basis for coordination in game theory and communication.

0.107.3 16.3 Paraconsistent Logic and Belief Revision

Paraconsistent logics are logics in which contradictions do not entail everything. In classical logic, from φ and $\neg\varphi$, any ψ is derivable (ex contradictione quodlibet). This makes classical reasoning fragile in the presence of inconsistent information.

A paraconsistent logic weakens the principle of explosion. The family of **LP** (Logic of Paradox) logics allows formulas to be both true and false simultaneously without destroying the entire inference system. This is motivated by real-world databases, legal reasoning, and temporal belief states in which an agent holds contradictory beliefs during a process of revising its world model.

Belief revision (Alchourrón-Gärdenfors-Makinson, AGM theory) addresses the coherent update of a belief set K upon receiving new information φ. Three operations are defined: **expansion** ($K + \varphi$), adding φ to the belief set; **contraction** ($K - \varphi$), removing φ while remaining as conservative as possible; and **revision** ($K * \varphi$), incorporating φ even if $\neg\varphi \in K$, using the Levi identity $K * \varphi = (K - \neg\varphi) + \varphi$. The AGM postulates for rational revision are elegant axioms characterising what any coherent belief revision operation must satisfy, and Grove's sphere semantics proves that AGM-rational revision operators correspond exactly to orderings of possible worlds by their epistemic distance from the current belief state — a connection to Lewis's counterfactual semantics.

0.108 16h. Formal Argumentation Theory

0.108.1 16h.1 Dung's Abstract Argumentation Frameworks

Abstract argumentation (Dung 1995) models reasoning as a conflict between arguments, abstracting away the internal structure of arguments to focus on the attack relations between them.

An **abstract argumentation framework** (AAF) is a pair $\mathcal{AF} = (\mathcal{A}, \mathcal{R})$ where $\mathcal{A}$ is a set of **arguments** and $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$ is the **attack relation**: $(a, b) \in \mathcal{R}$ means argument a attacks argument b.

A set $S \subseteq \mathcal{A}$ of arguments is:

- **Conflict-free**: no argument in S attacks another argument in S.
- **Defends** a (i.e., S is **acceptable** w.r.t. a): for every b attacking a, some $c \in S$ attacks b.

- An **admissible set**: conflict-free and defends all its members.
- A **complete extension**: admissible and contains every argument it defends.
- A **stable extension**: conflict-free and attacks every argument not in S.
- The **grounded extension**: the smallest complete extension (the most sceptical conclusion).
- A **preferred extension**: a maximal (by set inclusion) admissible set (the most credulous conclusions).

Theorem 0.108.1 (Dung 1995)**.** Every AAF has a unique grounded extension and at least one complete extension. Preferred extensions are complete extensions. Stable extensions are preferred extensions (but not vice versa in general).

0.108.2 16h.2 Probabilistic Argumentation

Probabilistic argumentation (Hunter and Thimm 2017) assigns probabilities to arguments, modelling uncertainty about which arguments are valid. A **probabilistic AAF** (PrAAF) assigns a probability $P(a) \in [0, 1]$ to each argument $a \in \mathcal{A}$ and extends the acceptance semantics to probabilistic statements.

The probability that a set S is an extension under semantics σ is:

$$P_\sigma(S) = \sum_{G \subseteq \mathcal{A}: S = \sigma(G)} \prod_{a \in G} P(a) \prod_{b \notin G} (1 - P(b)),$$

where $\sigma(G)$ is the extension of the subframework G under semantics σ. The probability that an argument a is **accepted** is $P^{\text{acc}}_\sigma(a) = \sum_{S: a \in S} P_\sigma(S)$.

Cognitive application. Argumentation provides a formal framework for modelling legal, ethical, and everyday reasoning involving conflicting considerations. A judicial decision process can be modelled as finding the preferred or grounded extension of an AAF where arguments represent legal claims and attacks represent counter-arguments. The probability assigned to each argument reflects the strength of evidence; the probabilistic extension semantics aggregates these into a probability of legal acceptance for each claim.

0.109 16g. Complexity of Nonmonotonic Reasoning

0.109.1 16g.1 Complexity Classes for Nonmonotonic Inference

The computational complexity of nonmonotonic reasoning systems is significantly higher than classical propositional logic (NP-complete). This has important implications for the cognitive plausibility of these formalisms.

For **Reiter's default logic** (Section 16e):

- **Consistency of an extension**: given a default theory Δ and a proposed extension E, is E consistent? This is in P (polynomial time) given E, since consistency of a propositional theory is co-NP, but checking the fixed-point condition is tractable for finite theories.
- **Extension existence**: does Δ have an extension? Σ_2^P-complete.

- **Brave reasoning** (does φ belong to some extension?): Σ_2^P-complete.
- **Cautious reasoning** (does φ belong to all extensions?): Π_2^P-complete.

For **circumscription** (McCarthy 1980): minimal-model reasoning is Π_2^P-complete.

For **Answer Set Programming** (stable model semantics):

- Deciding if a stable model exists: NP-complete.
- Deciding if φ is in all stable models: co-NP-complete.

These complexity results suggest that exact nonmonotonic reasoning is computationally intractable for large knowledge bases. The brain must use approximate nonmonotonic reasoning: heuristics that capture the most important defaults and exceptions without exhaustive search over all possible extensions.

0.109.2 16g.2 Approximate Nonmonotonic Reasoning

Several tractable approximations to full nonmonotonic reasoning exist:

Horn clause approximation. Restrict to Horn clauses (at most one positive literal per clause). Propositional Horn satisfiability is tractable (P), and Horn default logic is tractable for specific classes of defaults. Most practical knowledge bases (ontologies, rule-based systems) are designed to be tractable via Horn or near-Horn encodings.

Probabilistic approximation. Replace nonmonotonic inference with probabilistic inference: defaults become high-probability rules rather than absolute ones, and probabilistic consistency replaces logical consistency. Maximum Entropy inference (Section 13b) provides a principled bridge: the probability distribution that maximises entropy subject to the constraint that each default rule $\alpha | \sim \beta$ holds with probability $1 - \epsilon$ (for small ϵ) gives a tractable, graded version of nonmonotonic reasoning.

Case-based reasoning. Represent defaults as examples rather than rules: instead of "birds fly by default," store exemplars of (bird, flies) pairs and use similarity-based retrieval. Given a new case, retrieve the most similar stored case and inherit its properties. Case-based reasoning is empirically tractable (linear in the case base size) and cognitively plausible (humans often reason by analogy to remembered examples).

0.110 16f. Formal Epistemology: Belief Merging and Social Learning

0.110.1 16f.1 Belief Merging Operators

When multiple rational agents with possibly different beliefs must form a collective belief, **belief merging** operators aggregate their epistemic states. For a group of agents with belief sets $K_1, \ldots, K_n$ (deductively closed sets of propositional formulas), a merging operator Δ_μ takes a **profile** $(K_1, \ldots, K_n)$ and an **integrity constraint** μ (a formula that the result must satisfy) and returns a merged belief set:

$$\Delta_\mu(K_1, \ldots, K_n) = \text{a belief set satisfying } \mu \text{ that best approximates the agents' beliefs.}$$

Axioms for rational merging (Konieczny and Pino Pérez 2002):

1. **(IC0)**: $\Delta_\mu(\mathbf{E}) \models \mu$ (the result satisfies the constraint).

2. **(IC1)**: If μ is consistent with $\bigwedge_i K_i$, then $\Delta_\mu(\mathbf{E}) \equiv \bigwedge_i K_i \wedge \mu$.

3. **(IC2)**: If $K_i \equiv K_j$ for all i, j, then $\Delta_\mu(\mathbf{E}) \equiv K_1 \wedge \mu$ (unanimity).

4. **(IC3)**: $\Delta_\mu(\mathbf{E}) \cup \{\mu\} \not\vdash \bot$ (consistency with constraint).

The **distance-based merging** approach selects the model of μ that minimises the aggregate distance to the agents' preferred models:

$$\Delta_\mu(\mathbf{E}) = \arg\min_{\omega} \left(\omega \models \mu, \bigoplus_i d(\omega, K_i) \right),$$

where $d(\omega, K_i) = \min_{\omega' \models K_i} d_H(\omega, \omega')$ is the Hamming distance from ω to the nearest model of K_i, and $\bigoplus$ is an aggregation function (sum, max, or lexicographic ordering).

Bayesian belief merging. When beliefs are represented as probability distributions $P_1, \ldots, P_n$, the natural merging operators are:

- **Linear pooling**: $\bar{P} = \sum_i w_i P_i$ (a mixture of the agents' distributions).
- **Logarithmic pooling**: $\bar{P} \propto \prod_i P_i^{w_i}$ (the geometric mean of the agents' distributions, normalised).

Logarithmic pooling satisfies **external Bayesian rationality**: if all agents update from the same prior by Bayes' theorem on the same evidence, the pooled result is the same as updating the pooled prior. Linear pooling does not satisfy this property. For cognitive science, logarithmic pooling models the aggregation of beliefs in groups that share evidence (e.g., a jury or a scientific community).

0.110.2 16f.2 Social Learning and Opinion Dynamics

Social learning models how agents update their beliefs by observing others' actions or beliefs. The **DeGroot model** (DeGroot 1974) is a linear opinion update rule:

$$p_i^{(t+1)} = \sum_j T_{ij} p_j^{(t)},$$

where $p_i^{(t)}$ is agent i's belief (probability) at time t and T is a row-stochastic **trust matrix** ($T_{ij} \geq 0$, $\sum_j T_{ij} = 1$). The agent weights its own and others' beliefs by T_{ij}.

Theorem 0.110.1 (DeGroot consensus)**.** If the trust graph (directed graph with edge (i, j) iff $T_{ij} > 0$) is strongly connected and aperiodic, then all agents converge to the same consensus belief:

$$p_i^{(t)} \to p^* = \pi^\top \mathbf{p}^{(0)},$$

where π is the stationary distribution of the Markov chain with transition matrix T and $\mathbf{p}^{(0)}$ is the vector of initial beliefs.

The consensus is the stationary-weighted average of initial beliefs: more influential agents (higher π_i) have more weight in the final consensus. The model predicts that agents whose opinions are most sought-after (high in-degree) are most influential. This is a mathematical formalisation of the **opinion leader** phenomenon in social psychology.

Bayesian social learning (Banerjee 1992; Bikhchandani et al. 1992) is rational social learning where agents observe others' actions (not beliefs) and perform Bayesian updating. Under sequential decision-making, rational agents can **herd**: after enough agents have chosen action A, the public information in favour of A overwhelms any private signal in favour of B, and subsequent agents cascade to A even if A is suboptimal. This **information cascade** is a rational failure mode of social learning, arising because actions (discrete choices) are less informative than beliefs (continuous probabilities).

0.111 16e. Default Logic: Complete Formal Treatment

0.111.1 16e.1 Reiter's Default Logic

Default logic (Reiter 1980) extends first-order logic with **default rules** of the form:

$$\frac{\alpha(\mathbf{x}) : \beta_1(\mathbf{x}), \ldots, \beta_m(\mathbf{x})}{\gamma(\mathbf{x})},$$

read: "If $\alpha(\mathbf{x})$ is believed, and each of $\beta_1(\mathbf{x}), \ldots, \beta_m(\mathbf{x})$ is consistent with current beliefs, then conclude $\gamma(\mathbf{x})$." The formula α is the **prerequisite**, $\beta_1, \ldots, \beta_m$ are the **justifications**, and γ is the **consequent**.

A **default theory** is a pair $\Delta = (W, D)$ where W is a set of first-order formulas (the facts) and D is a set of default rules. An **extension** of Δ is a set E of formulas that is:

1. A deductively closed superset of W: $W \subseteq E$ and $E = \text{Th}(E)$.
2. Consistent: $E \nvdash \bot$.
3. Closed under applicable defaults: if $\delta = \frac{\alpha:\beta_1,\ldots,\beta_m}{\gamma} \in D$ and $\alpha \in E$ and $\neg\beta_i \notin E$ for all i, then $\gamma \in E$.

Extensions are the "maximally rational" sets of beliefs derivable from Δ. Multiple extensions represent alternative consistent belief states; a unique extension corresponds to a well-determined belief state.

Example (bird flying). $W = \{\text{Bird(Tweety)}\}$, $D = \{d_1 = \frac{\text{Bird}(x):\text{Flies}(x)}{\text{Flies}(x)}\}$. The unique extension is $E = \text{Th}(W \cup \{\text{Flies(Tweety)}\})$: birds typically fly, so Tweety flies by default.

Adding $W' = W \cup \{\text{Penguin(Tweety)}\}$ and $d_2 = \frac{\text{Penguin}(x):\neg\text{Flies}(x)}{\neg\text{Flies}(x)}$: now d_1 and d_2 conflict. Default logic resolves by **specificity**: d_2 is triggered (Penguin is more specific than Bird), giving extension $E = \text{Th}(W' \cup \{\neg\text{Flies(Tweety)}\})$.

0.111.2 16e.2 The Fixed-Point Equations

Extensions can be characterised by a fixed-point equation. For a default theory (W, D), define the operator $\Gamma : 2^L \to 2^L$ mapping a set of formulas S to:

$$\Gamma(S) = \text{smallest deductively closed set} E \text{ s.t.:}$$

$$W \subseteq E, \text{ and for all } \delta = \frac{\alpha : \beta_1, \ldots, \beta_m}{\gamma} \in D, \text{ if } \alpha \in E \text{ and } \neg\beta_i \notin S \text{ for all } i, \text{ then } \gamma \in E.$$

Then E is an extension iff $\Gamma(E) = E$ (fixed point). Computing all fixed points is Σ_2^P-complete (the second level of the polynomial hierarchy), making default reasoning strictly harder than classical satisfiability

(NP-complete) but tractable in practice for small $|D|$.

0.111.3 16e.3 Cognitive Applications: Typicality and Inheritance

Default logic naturally models **typicality reasoning** and **inheritance with exceptions** in cognitive science. A simple taxonomic hierarchy:

- $W = \{\text{Animal}(\text{Tweety}), \text{Bird}(\text{Tweety}), \text{Penguin}(\text{Tweety})\}$
- $d_1 = \frac{\text{Animal}(x):\text{HasLegs}(x)}{\text{HasLegs}(x)}$ (animals typically have legs)
- $d_2 = \frac{\text{Bird}(x):\text{Flies}(x)}{\text{Flies}(x)}$ (birds typically fly)
- $d_3 = \frac{\text{Penguin}(x):\neg\text{Flies}(x)}{\neg\text{Flies}(x)}$ (penguins typically don't fly)

Priority among defaults is specified by the **specificity principle**: $d_3 \succ d_2 \succ d_1$ (more specific takes priority). The unique extension contains HasLegs(Tweety) (from d_1, no conflict) and $\neg$Flies(Tweety) (from d_3, overriding d_2).

The psychological correspondence: typicality judgments in humans exhibit a **graded specificity effect**. More specific categories override more general ones in default inference (a penguin is not a typical bird in the flight respect, but is a typical bird in other respects). The specificity ordering in default logic formalises the psychological notion of **nested exceptions** in taxonomic knowledge.

0.112 16b. AGM Belief Revision: The Postulates and Grove's Theorem

0.112.1 16b.1 The Six AGM Postulates for Revision

The AGM theory (Alchourrón, Gärdenfors, Makinson 1985) characterises rational belief revision by eight postulates. For revision operator $*$ and belief set K (a deductively closed set of sentences):

(K*1) **Closure:** $K * \varphi$ is a belief set (deductively closed).

(K*2) **Success:** $\varphi \in K * \varphi$ (the new information is believed).

(K*3) **Inclusion:** $K * \varphi \subseteq \overline{K + \varphi}$ (revision does not add more than expansion).

(K*4) **Vacuity:** If $\neg\varphi \notin K$, then $K + \varphi \subseteq K * \varphi$ (if φ is consistent with K, revision equals expansion).

(K*5) **Consistency:** $K * \varphi \nvdash \bot$ if φ is consistent.

(K*6) **Extensionality:** If $\vdash \varphi \leftrightarrow \psi$, then $K * \varphi = K * \psi$.

(K*7) **Superexpansion:** $K * (\varphi \wedge \psi) \subseteq \overline{(K * \varphi) + \psi}$.

(K*8) **Subexpansion:** If $\neg\psi \notin K * \varphi$, then $\overline{(K * \varphi) + \psi} \subseteq K * (\varphi \wedge \psi)$.

Postulates (K*1)–(K*6) are the **basic postulates** sufficient for unique characterisation; (K*7) and (K*8) are the supplementary postulates ensuring coherent iterated revision.

Theorem 0.112.1 (Grove 1988)**.** $*$ satisfies the basic AGM postulates (K*1)–(K*6) if and only if there exists a **system of spheres** $\mathcal{S} = \{S_\alpha\}_\alpha$ (a nested family of sets of possible worlds, with S_0 being the

worlds consistent with K) such that:

$$K * \varphi = \text{Th}(S_{\alpha_0} \cap [\![\varphi]\!]),$$

where α_0 is the smallest index such that $S_{\alpha_0} \cap [\![\varphi]\!] \neq \emptyset$ and $[\![\varphi]\!]$ is the set of worlds satisfying φ.

The system of spheres is a **plausibility ordering** on possible worlds: worlds closer to the centre (lower α) are considered more plausible. Revision by φ picks the most plausible worlds satisfying φ. This is Lewis's semantics for counterfactuals, applied to belief revision: "What would I believe if I learned φ?" means "What is true in the closest worlds where φ holds?"

0.112.2 16b.2 Ranked Entailment and Spohn's Ordinal Conditional Functions

Ranking theory (Spohn 1988) provides a dynamic alternative to the static AGM framework, using **ordinal conditional functions** (OCFs) $\kappa : \Omega \to \mathbb{N} \cup \{\infty\}$ as a representation of degrees of disbelief. The rank $\kappa(\omega)$ is the degree of disbelief in world ω; $\kappa(\omega) = 0$ means ω is not disbelieved; $\kappa(\omega) = \infty$ means ω is absolutely disbelieved.

For events $A \subseteq \Omega$: $\kappa(A) = \min_{\omega \in A} \kappa(\omega)$ (the rank of the most plausible world in A). The belief in φ (i.e., $[\![\varphi]\!]$) is:

$$\text{believe}(\varphi) \iff \kappa(\neg\varphi) > 0, \qquad \text{i.e., } \kappa(\neg[\![\varphi]\!]) > 0.$$

Conditioning in ranking theory: $\kappa(\omega \mid A) = \kappa(\omega) - \kappa(A)$ for $\omega \in A$. This is the ordinal analogue of Bayesian conditioning. The dynamics of rank update capture **conditionalization** (learning A with certainty), **imaging** (moving probability mass to nearest A-worlds), and **Jeffrey update** (partial shifts in rank).

Ranking theory provides a discrete, computationally tractable alternative to the full probabilistic Bayesian framework, appropriate for modelling qualitative belief change: the passage from "I believed P" to "I now believe not-P" after learning new evidence.

0.113 16c. Dynamic Epistemic Logic

0.113.1 16c.1 Public Announcement Logic

Dynamic Epistemic Logic (DEL; Plaza 1989; Gerbrandy and Groeneveld 1997; van Ditmarsch et al., 2007) extends static epistemic logic with operators that model the effect of information-changing events (announcements, eavesdropping, deception) on agents' knowledge.

The simplest extension is **Public Announcement Logic** (PAL). An announcement "φ!" ("it is publicly announced that φ") changes the model: every agent's accessibility relation is restricted to the worlds where φ holds. Formally, after announcement $\varphi!$ in model $\mathcal{M} = (W, R_1, \ldots, R_n, V)$, the new model is $\mathcal{M}|\varphi = (W^\varphi, R_1^\varphi, \ldots, R_n^\varphi, V^\varphi)$ where:

$$W^\varphi = \{w \in W : \mathcal{M}, w \vDash \varphi\}, \qquad R_i^\varphi = R_i \cap (W^\varphi \times W^\varphi).$$

The **reduction axiom** for PAL expresses the effect of announcement on knowledge:

$$[\varphi!]\, K_i\psi \Leftrightarrow (\varphi \Rightarrow K_i(\varphi \Rightarrow [\varphi!]\psi)).$$

This says: after a truthful announcement of φ, agent i knows ψ iff (given that φ is true) i already knew that: if φ then after the announcement of φ, ψ would hold. The formula on the right has strictly fewer announcement operators (the announcement depth decreases), so this reduction axiom allows any PAL formula to be translated into a pure epistemic logic formula in finitely many steps.

Example 0.113.1 (Muddy Children Puzzle)**.** n children stand in a circle; some have muddy faces. Each can see all others but not themselves. The father announces "At least one child has a muddy face," then asks repeatedly "Do you know whether your face is muddy?" until children who know raise their hand.

With k muddy children, exactly k rounds of questioning are needed before all muddy children simultaneously raise their hands. The first announcement updates common knowledge: previously, a child with $k-1$ muddy others could not distinguish between the world where they are muddy (total k) and the world where they are not (total $k-1$). The public announcement adds the information that $k-1 \geq 1$ (there is at least one muddy child), which propagates through iterated reasoning.

Formally, let $m_i \in \{0,1\}$ be the state of child i's face. After the announcement $\exists i : m_i = 1$, the model restricts to worlds where at least one child is muddy. After j rounds of questions (with no hands raised), the model further restricts: every child knows that no child with fewer than j muddy peers knows their own state. By induction, after k rounds, the muddy children can deduce their state. □

0.113.2 16c.2 Action Models and Full DEL

Full DEL handles more complex epistemic events (lying, eavesdropping, partial announcements) via **action models** $\mathcal{A} = (E, Q_1, \ldots, Q_n, \text{pre}, \text{post})$ where E is a set of epistemic actions, Q_i is agent i's uncertainty over actions, $\text{pre} : E \to \mathcal{L}$ specifies the precondition of each action, and $\text{post} : E \to (P \to \mathcal{L})$ specifies its postconditions on atomic propositions.

The **product update** of model $\mathcal{M}$ with action model $\mathcal{A}$ at event e is the model $\mathcal{M} \otimes \mathcal{A}$ with:

$$W' = \{(w, e') : w \in W, e' \in E, \mathcal{M}, w \vDash \text{pre}(e')\},$$

$$R_i' = \{((w,e),(w',e')) : (w,w') \in R_i, (e,e') \in Q_i\},$$

$$V'(p) = \{(w,e') : \mathcal{M}, w \vDash \text{post}(e')(p)\}.$$

The product update models the simultaneous change of agents' worlds (the postconditions) and knowledge (the accessibility relation restricted to action uncertainty). Recursive announcements, lies, and higher-order deception can be modelled by composing action models.

Cognitive relevance: DEL provides the mathematical framework for modelling Theory of Mind in multi-agent settings — the capacity to reason about what other agents know, believe, and plan. Developmental evidence shows that children acquire full second-order Theory of Mind (reasoning about what A knows about what B knows) around age 4–5, correlating with the ability to pass False Belief tasks. DEL's semantics makes precise the computational demands of k-th order epistemic reasoning: it requires maintaining and updating a k-th order Kripke model, whose size grows exponentially in k.

0.113.3 16c.3 Conditional Logic and Non-Monotonic Consequence Relations

Conditional logic (Lewis 1973; Stalnaker 1968) formalises counterfactual and indicative conditionals as operators in modal logic. The **variably strict** conditional $\varphi \Rightarrow \psi$ ("if φ were the case, then ψ would be the case") is evaluated as:

$$\mathcal{M}, w \vDash \varphi \Rightarrow \psi \iff \text{the closest world to } w \text{ where } \varphi \text{ holds satisfies } \psi,$$

where "closeness" is given by a total preorder $\leq_w$ on worlds (worlds closer to w are more similar to w). The semantics is the sphere semantics of Grove (1988), connecting directly to the AGM framework of Section 16b.

The **Ramsey Test** connection: $\varphi \Rightarrow \psi \in K$ (the belief set) iff $\psi \in K * \varphi$ (revision of K by φ yields ψ). This links conditional logic to belief revision: the belief in a conditional is equivalent to the result of hypothetically revising by the antecedent and checking the consequent.

Preferential entailment (Shoham 1988; Kraus, Lehmann, Magidor 1990) provides a non-monotonic consequence relation $\alpha \mathrel{|\!\sim} \beta$ ("α typically entails β") via a preference order on models: $\alpha \mathrel{|\!\sim} \beta$ iff every "preferred" α-model is also a β-model. The KLM postulates characterise rational preferential entailment systems:

- **Reflexivity:** $\alpha \mathrel{|\!\sim} \alpha$.
- **Left equivalence:** If $\vdash \alpha \leftrightarrow \beta$ and $\alpha \mathrel{|\!\sim} \gamma$, then $\beta \mathrel{|\!\sim} \gamma$.
- **Right weakening:** If $\alpha \mathrel{|\!\sim} \beta$ and $\vdash \beta \Rightarrow \gamma$, then $\alpha \mathrel{|\!\sim} \gamma$.
- **Cut:** If $\alpha \mathrel{|\!\sim} \beta$ and $\alpha \wedge \beta \mathrel{|\!\sim} \gamma$, then $\alpha \mathrel{|\!\sim} \gamma$.
- **Cautious monotony:** If $\alpha \mathrel{|\!\sim} \beta$ and $\alpha \mathrel{|\!\sim} \gamma$, then $\alpha \wedge \beta \mathrel{|\!\sim} \gamma$.
- **Or:** If $\alpha \mathrel{|\!\sim} \gamma$ and $\beta \mathrel{|\!\sim} \gamma$, then $\alpha \vee \beta \mathrel{|\!\sim} \gamma$.

These are strictly weaker than classical logic (which satisfies all of these plus monotonicity: if $\alpha \vdash \gamma$, then $\alpha \wedge \beta \vdash \gamma$). The KLM systems are the standard framework for formalising "typicality" reasoning in cognitive science and law.

Section IX

Section IX — Probabilistic Programming

0.114 17. Programs as Models

Probabilistic programming is the discipline of expressing probabilistic generative models as programs and performing inference over the program's execution. A probabilistic program is a syntactic object — source code in a Turing-complete language extended with random sampling primitives — and its semantics is a probability distribution over execution traces.

The fundamental primitive is a `sample` statement: `x = sample(dist)` draws a value from `dist` and binds it to `x`. The `observe` (or `condition`) statement constrains the trace: `observe(dist, value)` reweights the current trace so that the probability of generating that trace is proportional to `dist.pdf(value)`.

A probabilistic program implicitly defines a **generative process**: the distribution over traces induced by running the program with all sampling operations drawing independently from their distributions. The posterior distribution, conditioned on observed data, is the distribution over traces consistent with the observations. Inference in a probabilistic program is the computation of (summaries of) this posterior.

0.115 18. Measure-Theoretic Semantics

The denotational semantics of probabilistic programs is measure-theoretic. A program P in a language $\mathcal{L}$ with measurable state space $\mathcal{S}$ denotes a **sub-probability kernel** $[\![P]\!] : \mathcal{S} \to \mathcal{M}(\mathcal{S})$, where $\mathcal{M}(\mathcal{S})$ is the set of sub-probability measures on $\mathcal{S}$. Sub-probabilities allow for the possibility that programs may not terminate (the total weight $[\![P]\!](\sigma)(S) \leq 1$ reflects the probability of termination).

For a sequential composition $P; Q$, the semantics is:

$$[\![P; Q]\!](\sigma)(A) = \int_{\mathcal{S}} [\![Q]\!](\sigma')(A)\, [\![P]\!](\sigma)(d\sigma').$$

This is the Kleisli composition of sub-probability kernels, which forms a category — the **Kleisli category** of the sub-probability monad. The categorical structure of probabilistic programs is an active area connecting programming language theory, category theory, and cognitive modelling.

Universal probabilistic programming languages such as Church, Venture, Stan, and Pyro provide this expressive power in practice. A Church program can express any computable probability distribution, including distributions over programs themselves (higher-order probabilistic programs). This makes probabilistic programming a language for **Bayesian cognitive modelling**: expressing cognitive theories as generative programs and fitting them to behavioural data by posterior inference.

0.115.1 18.1 Inference Algorithms for Probabilistic Programs

Computing the posterior in a probabilistic program requires inference algorithms that are agnostic to the specific program structure. Three families of methods have been developed.

Importance sampling computes the expectation of a function f under the posterior by weighting traces from the prior by their likelihood:

$$\mathbb{E}[f(x) \mid \text{obs}] \approx \frac{\sum_{i=1}^{N} w_i f(x_i)}{\sum_{i=1}^{N} w_i}, \quad w_i = \frac{p(\text{obs} \mid x_i)}{q(x_i)},$$

where $x_i \sim q$ are traces sampled from a proposal and w_i are importance weights. Importance sampling is unbiased but inefficient in high dimensions (weight degeneracy).

Sequential Monte Carlo (particle filtering) maintains a population of weighted traces ("particles") and resamples them at each observation, correcting the weights as new evidence arrives. It is the inference algorithm underlying the hypothesis that the brain maintains a population code of probability distributions rather than point estimates.

Hamiltonian Monte Carlo (HMC), used in Stan and related systems, exploits gradient information to propose efficient moves in continuous latent space. It augments the target distribution with auxiliary momentum variables and simulates Hamiltonian dynamics to make large, correlated moves. The No-U-Turn Sampler (NUTS) adapts the trajectory length automatically, making HMC practical without manual tuning.

0.116 16d. Quantum Logic and Paraconsistent Reasoning

0.116.1 16d.1 The Birkhoff-von Neumann Lattice

Quantum logic (Birkhoff and von Neumann 1936) arises from the observation that the lattice of closed subspaces of a Hilbert space $\mathcal{H}$ does not satisfy the classical distributive law. For subspaces $A, B, C \subseteq \mathcal{H}$:

$$A \cap (B + C) \neq (A \cap B) + (A \cap C) \quad \text{in general,}$$

where $+$ denotes the closed linear span (the quantum-logical "or"). The failure of distributivity is the mathematical signature of quantum interference: the "or" of two quantum states is not simply their union but their superposition, and the projection onto a conjunction does not distribute over disjunctions.

The lattice $\mathcal{L}(\mathcal{H})$ of closed subspaces (with operations $A \wedge B = A \cap B$, $A \vee B = \overline{A + B}$, $A^\perp = A$'s orthogonal complement) is an **orthomodular lattice**: it satisfies the **orthomodular law** $A \leq B \Rightarrow B = A \vee (A^\perp \wedge B)$ instead of the distributive law. Orthomodularity is strictly weaker than distributivity; distributive orthomodular lattices are exactly the Boolean algebras (classical logic).

Cognitive applications. Busemeyer and Bruza (2012) argue that human judgment and decision-making violate classical probability theory in ways that are exactly predicted by quantum probability theory (probability on orthomodular lattices). The **conjunction fallacy** (judging "Linda is a bank teller and a feminist" more probable than "Linda is a bank teller") and the **order effect** (judgment probabilities depend on question order) are naturally explained by quantum probability's non-commutativity: $P(A \wedge B) \neq P(B \wedge A)$ in quantum probability, and order of measurement matters.

0.116.2 16d.2 Paraconsistent Logic

Paraconsistent logic rejects the classical **ex contradictione quodlibet** (ECQ): $A \wedge \neg A \vdash B$ for any B ("from a contradiction, anything follows"). Classical logic is **explosive**: a single contradiction trivialises the entire system. Paraconsistent logics are designed to be "tolerant" of contradictions, allowing contradictory information to coexist without entailing everything.

LP (Logic of Paradox; Priest 1979) is a three-valued logic with truth values $\{T, B, F\}$ where B means "both true and false." The semantics:

- $\neg A$ is T if A is F, F if A is T, B if A is B.
- $A \wedge B$ is T iff both are T (or both B with at least one T); F iff at least one is F and neither is B; B otherwise.

The designated truth values (those that make a formula "true enough to assert") are $\{T, B\}$. ECQ fails: $A \wedge \neg A$ evaluates to B (not T), so it does not derive B for arbitrary formulas.

For cognitive science, paraconsistent logic models **belief revision under inconsistency**: a rational agent who receives contradictory evidence need not abandon all beliefs but can maintain a coherent (if paraconsistent) state of knowledge until the contradiction is resolved. Legal reasoning, database management, and moral reasoning frequently involve temporary or partial inconsistency that classical logic handles poorly.

0.116.3 16d.3 Probabilistic Logic and Markov Logic Networks

Markov Logic Networks (MLNs; Richardson and Domingos 2006) provide a probabilistic extension of first-order logic that combines the expressiveness of logic with the uncertainty quantification of probability theory.

An MLN is a set of **weighted first-order formulas** $\{(F_i, w_i)\}$. Given a set of constants $\mathcal{C}$, the MLN defines a Markov network over the **ground atoms** (all instantiations of the predicates with constants from $\mathcal{C}$). The probability of a world (truth assignment to all ground atoms) is:

$$P(X = x) = \frac{1}{Z} \exp\left(\sum_i w_i n_i(x)\right),$$

where $n_i(x)$ is the number of true groundings of formula F_i in world x. This is a log-linear model where the features are the ground-truth counts of logical formulas.

The weight w_i encodes the strength of the constraint: $w_i = \infty$ means F_i is a hard constraint (must hold in every world); $w_i > 0$ means F_i's truth increases the world's probability; $w_i < 0$ means its truth decreases it. Classical first-order logic corresponds to the case where all weights are ∞.

Inference in MLNs (computing $P(\text{query}|\text{evidence})$) is performed by grounding the MLN for the relevant constants and running standard probabilistic inference (MCMC, belief propagation) on the resulting Markov network. Learning the weights from data is performed by maximising the conditional log-likelihood, analogous to logistic regression at the level of logical formulas.

0.117 18d. Continuation-Passing Semantics and Computational Effects

0.117.1 18d.1 The CPS Transform

The semantics of probabilistic programs is naturally expressed in **continuation-passing style** (CPS). In CPS, each function takes an extra argument: the **continuation** k, a function representing "the rest of the computation." Rather than returning a value, the function passes its result to k.

For a deterministic expression e that computes value v, the CPS transform is:

$$\text{CPS}(e) = \lambda k.k(v).$$

For a probabilistic expression $\texttt{sample}(d)$ that samples from distribution d:

$$\text{CPS}(\texttt{sample}(d)) = \lambda k. \int k(x)\, d\mu(x),$$

where the integral marginalises over the sampled value. For a conditional expression $\texttt{condition}(b)$ (observe that boolean b is true):

$$\text{CPS}(\texttt{condition}(b)) = \lambda k. b \cdot k(\text{true}),$$

which multiplies the measure by the Boolean indicator of b (zero weight if b is false, unit weight if true). This is the measure-theoretic encoding of conditioning.

The CPS transform composites into the **Giry monad** (Section 18b.3): the continuation k has type $A \to \mathcal{G}(R)$ (a Kleisli arrow), and the probabilistic program has type $\mathcal{G}(R)$ (a distribution over results). The bind operation of the Giry monad corresponds to sequential composition in CPS:

$$(m \gg\!= f)k = m(\lambda x. f(x)k) = \int f(x)(k)\, d\mu_m(x).$$

0.117.2 18d.2 Trace-Based Inference: Sequential Monte Carlo

For general probabilistic programs with complex control flow (recursion, branching), the posterior distribution over program executions (traces) requires specialised inference algorithms. **Sequential Monte Carlo** (SMC) is the most widely applicable exact sampler for structured generative models.

An execution **trace** of a probabilistic program is a record of all random choices made during execution: $\tau = (c_1, v_1), (c_2, v_2), \ldots, (c_T, v_T)$ where c_t is the choice point and v_t is the sampled value. The weight of a trace is the product of the likelihood factors encountered:

$$w(\tau) = \prod_{t:\text{observed at } t} p(\text{obs}_t | v_t) \cdot \prod_{t:\text{conditioned}} \mathbf{1}[\text{condition}_t(\tau)],$$

where the first product is over `observe` statements (soft constraints) and the second over `condition` statements (hard constraints).

SMC maintains a population of N particles (weighted traces), propagates them through the program, and resamples at `observe` points. The algorithm:

1. **Initialise**: Sample N traces from the prior $\tau^{(i)} \sim p(\tau)$.
2. **Propagate**: Run each trace forward until the next `observe` statement.
3. **Weight**: Compute $w_t^{(i)} = p(\text{obs}_t | v_t^{(i)})$ for each particle.
4. **Resample**: Sample with replacement from the population, weighted by $w_t^{(i)}$.
5. **Repeat** until all observations are processed.

The SMC estimate of the posterior expectation $\mathbb{E}[f(\tau)|D]$ is $\frac{1}{N}\sum_i f(\tau^{(i)})$ after the final resampling. SMC is consistent (converges as $N \to \infty$) and can handle discrete variables, branching programs, and programs with variable-length traces (unlike MCMC, which requires fixed-dimensional state spaces).

The **evidence** (marginal likelihood $p(D)$) is estimated as the product of the normalising constants at each resampling step:

$$\hat{p}(D) = \prod_{t} \frac{1}{N} \sum_{i} w_t^{(i)}.$$

This estimate is unbiased and converges at rate $O(1/\sqrt{N})$ by the central limit theorem applied to the product of sample means.

0.118 18b. The Church Language and Probabilistic Lambda Calculus

0.118.1 18b.1 Church as a Universal Generative Language

Church (Goodman et al., 2008) is a probabilistic programming language built on Scheme (a dialect of Lisp) with two additional primitives: `(sample distribution)` and `(condition event)`. Its syntax is that of the lambda calculus augmented with stochastic primitives, making it Turing-complete as a probabilistic language.

A Church program is a **generative process**: a procedural description of how to produce data from latent variables. For example, a simple model of coin bias:

```
(define fair-coin? (lambda () (flip 0.5)))
(define biased-coin? (lambda () (flip 0.9)))
(define coin-type (if (flip 0.5) 'fair 'biased))
(define toss
  (if (eq? coin-type 'fair)
      (fair-coin?)
      (biased-coin?)))
(condition (eq? toss #t))
(query coin-type)
```

The `condition` statement constrains the distribution to only those execution traces where `toss` is `true`. The `query` computes the posterior distribution over `coin-type`: by Bayes' theorem, $p(\text{biased}|\text{toss} = \text{H}) \propto 0.5 \cdot 0.9 = 0.45$ and $p(\text{fair}|\text{toss} = \text{H}) \propto 0.5 \cdot 0.5 = 0.25$, giving $p(\text{biased}|\text{H}) = 0.45/0.70 \approx 0.643$.

The expressive power of Church includes recursion (higher-order stochastic processes), higher-order probabilistic functions (distributions over distributions), and unbounded computation. A **non-parametric mixture model** is expressible as a Church program using the *Chinese Restaurant Process*:

```
(define CRP
  (lambda (alpha)
    (let loop ((tables '()) (counts '()))
      (let* ((probs (append counts (list alpha)))
             (table (multinomial probs)))
        (if (= table (length counts))
            (begin (set! tables (append tables (list (gensym))))
                   (set! counts (append counts (list 1)))
                   (last tables))
            (begin (set! counts (update counts table +1))
                   (list-ref tables table)))))))
```

This generates an infinite sequence of cluster assignments from the CRP, which is the stick-breaking representation of the Dirichlet process. The corresponding Bayesian nonparametric model — a Dirichlet process mixture — is directly executable, and posterior inference (which cluster does each observation belong to?) is performed by the Church inference engine.

0.118.2 18b.2 WebPPL: Probabilistic Programming in JavaScript

WebPPL (Goodman and Stuhlmüller, 2014) is a probabilistic language implemented as a subset of JavaScript, enabling probabilistic programs to run in web browsers and interact with neural networks and data visualisation tools. Its functional style mirrors Church but compiles to continuation-passing style (CPS) to enable enumeration-based inference.

A key feature of WebPPL is its support for multiple inference backends on the same probabilistic program:

```
var model = function() {
  var theta = beta(1, 1);
  var likelihood = function(k, n) { binomial(theta, n) == k; };
  condition(likelihood(7, 10));
  return theta;
};
// Exact enumeration (discrete):
var posterior = Enumerate(model);
// MCMC (continuous):
var posterior = Infer({method: 'MCMC', samples: 10000}, model);
// Variational:
var posterior = Infer({method: 'optimize', samples: 1000}, model);
```

The same program can be evaluated under different inference algorithms without modification. This **inference-independent semantics** is the key feature of probabilistic programming: the program specifies what is to be inferred (the generative model and likelihood), not how (the inference algorithm).

0.118.3 18b.3 Denotational Semantics: The Probability Monad

The denotational semantics of probabilistic programs is given by the **probability monad** (Giry 1982; Lawvere 1962). The Giry monad $\mathcal{G}$ assigns to each measurable space X the space $\mathcal{G}X$ of probability measures on X, and to each measurable function $f : X \to Y$ a function $\mathcal{G}f : \mathcal{G}X \to \mathcal{G}Y$ defined by pushforward.

The monad structure provides:

- Unit $\eta_X : X \to \mathcal{G}X$, $\eta_X(x) = \delta_x$ (the Dirac measure at x): deterministic programs are probabilistic programs that return point masses.
- Multiplication $\mu_X : \mathcal{G}\mathcal{G}X \to \mathcal{G}X$, $\mu_X(\Pi)(A) = \int \pi(A)\,\Pi(d\pi)$: sequential composition marginalises over intermediate distributions.

A **Kleisli arrow** $f : X \to \mathcal{G}Y$ is a probabilistic function from X to distributions over Y. Sequential composition of probabilistic programs corresponds to Kleisli composition:

$$(f \star g)(x)(C) = \int \mathcal{G}g(f(x))(C) = \int_y g(y)(C)\, f(x)(dy).$$

This is exactly the law of total probability: the probability of C after executing f then g from initial state x.

For **conditioning**, the semantics requires **disintegration**: given a joint distribution μ on $X \times Y$ and a conditioning event $Y = y$, the conditional distribution $\mu_{X|Y=y}$ exists as a regular conditional distribution under mild measurability conditions (the existence of a disintegration). Disintegration is the measure-theoretic version of Bayesian conditioning, and its existence (which fails for some pathological spaces) is the technical condition ensuring that conditioning on events of measure zero is well-defined.

0.119 18g. Probabilistic Graphical Models as Cognitive Architectures

0.119.1 18g.1 Cognitive Architectures as Factor Graphs

Factor graphs (Section 14.1) provide a unifying representation for cognitive architectures. Different cognitive processes correspond to different factor structures:

Perception: a hierarchical factor graph where factors encode the causal relationships between latent causes (objects, scenes, properties) and sensory observations. Inference in the factor graph is perception (computing the posterior over causes given observations). The architecture of the factor graph encodes the prior knowledge about causal structure.

Memory: a factor graph with long-range temporal connections encoding the dependence of current states on past states. The temporal depth of these connections determines the working memory span. Hebbian plasticity updates the factor weights based on the co-occurrence of states in the training data.

Language: a factor graph over words, syntactic categories, and semantic representations. The CRF (Conditional Random Field) model of language (Lafferty et al., 2001) is a factor graph: $p(y|x) \propto \exp(\sum_i \sum_j \theta_{ij} f_{ij}(y_i, y_j, x))$ where y are the output sequence (e.g., POS tags) and x is the input sequence. The factors f_{ij} encode compatibility between adjacent output labels and between output labels and input words.

0.119.2 18g.2 Probabilistic Soft Logic

Probabilistic Soft Logic (PSL; Bach et al., 2017) is a framework for probabilistic reasoning over relational domains using the Hinge-Loss Markov Random Field (HL-MRF). PSL extends first-order logic with:

- **Soft truth values** $\in [0, 1]$ (grounded atoms are probabilities, not Boolean).
- **Lukasiewicz logic** for connectives: $A \wedge B = \max(A + B - 1, 0)$, $A \vee B = \min(A + B, 1)$, $\neg A = 1 - A$.
- **Weighted rules** mapping to hinge-loss potentials.

A PSL program consists of weighted rules like:

```
5.0: Knows(A,B) & Likes(A,C) -> Likes(B,C)  # transitive recommendations
3.0: Dislikes(A,C) -> !Likes(A,C)            # inconsistency penalty
```

The MAP inference problem is a convex quadratic program (for HL-MRFs), solvable efficiently with ADMM (Alternating Direction Method of Multipliers) or coordinate descent. PSL scales to millions of ground atoms, making it practical for large-scale knowledge graphs and social network reasoning.

Cognitive application. PSL provides a tractable implementation of the "theory of mind" computations required for social cognition: given soft evidence about relationships, preferences, and beliefs, PSL can efficiently compute the most probable assignment of mental states to agents. The soft truth values naturally handle the graded, uncertain nature of social knowledge.

0.120 18f. Measure Theory and Probabilistic Program Semantics

0.120.1 18f.1 The Radón-Nikodym Theorem and Conditioning

The mathematical foundation of conditioning in probabilistic programs is the **Radon-Nikodym theorem**:

Theorem 0.120.1 (Radon-Nikodym)**.** Let μ and ν be σ-finite measures on a measurable space $(\Omega, \mathcal{F})$ with $\nu \ll \mu$ (ν absolutely continuous w.r.t. μ). Then there exists a unique (up to μ-a.e.) measurable function $f : \Omega \to [0, \infty)$ such that $\nu(A) = \int_A f \, d\mu$ for all $A \in \mathcal{F}$.

The function $f = d\nu/d\mu$ is the **Radon-Nikodym derivative** (or density of ν w.r.t. μ). For probability measures, f is the density: $P(A) = \int_A p(x)\, d\mu(x)$ where μ is the reference measure (Lebesgue for continuous, counting for discrete).

Conditioning and disintegration. For a joint distribution $P(X, Y)$ and a conditioning event $Y = y$, the conditional distribution $P(X|Y = y)$ is a **disintegration** of P: a family of probability measures $\{P_y : y \in \mathcal{Y}\}$ such that $P(A \times B) = \int_B P_y(A)\, P_Y(dy)$ for all measurable $A \subseteq \mathcal{X}$, $B \subseteq \mathcal{Y}$.

Disintegration exists uniquely (a.e.) when the spaces are standard Borel spaces (Polish spaces with their Borel σ-algebras). This existence theorem underpins the validity of conditioning in probabilistic programs: the `observe` statement is semantically a disintegration, and its well-posedness follows from the disintegration theorem.

0.120.2 18f.2 The Lebesgue-Stieltjes Integral and Mixed Distributions

Many probabilistic programs produce **mixed distributions**: distributions with both continuous and discrete components. For example, a program that flips a coin and returns 0 if tails or a standard normal sample if heads:

```
def mixed():
    if bernoulli(0.5):
        return normal(0, 1)
    else:
        return 0
```

This program produces a distribution P that is neither absolutely continuous w.r.t. Lebesgue measure (it has an atom at 0) nor purely discrete (it has a continuous component on $\mathbb{R}$).

The correct mathematical framework for mixed distributions is the **Lebesgue-Stieltjes integral** with re-

spect to the **cumulative distribution function** (CDF) F:

$$\mathbb{E}[g(X)] = \int g(x)\, dF(x) = \int_{\{0\}} g(0)\, P(\{0\}) + \int_{\mathbb{R}} g(x)\, p_c(x)\, dx,$$

where p_c is the density of the absolutely continuous part of P (by the Lebesgue decomposition $P = P_{\text{disc}} + P_{\text{ac}} + P_{\text{sing}}$). The Lebesgue-Stieltjes integral unifies discrete and continuous expectations.

For probabilistic programs, the semantic domain must accommodate mixed distributions. The **s-finite kernel** framework (Staton 2017) provides a clean categorical semantics for probabilistic programs with conditioning and mixed distributions, avoiding the technical difficulties of working with improper measures.

0.120.3 18f.3 Importance Weights and the Likelihood Weighting Algorithm

Likelihood weighting (Fung and Chang 1989; Shachter and Peot 1990) is the simplest correct approximate inference algorithm for probabilistic programs with soft evidence (observe statements). The algorithm:

1. Sample a trace $\tau \sim p(\tau)$ by running the program forward without conditioning.
2. Compute the weight $w(\tau) = \prod_{\text{observe}(f_i(\tau))} f_i(\tau)$ (the product of all observed likelihood factors along the trace).
3. The posterior expectation is estimated as $\hat{\mathbb{E}}[g] = \sum_i w_i g(\tau_i) / \sum_i w_i$ (self-normalised importance sampling).

Likelihood weighting is unbiased (self-normalised estimator converges to the correct posterior expectation) but suffers from **weight degeneracy** when the observations are highly informative: most samples have near-zero weight and contribute little to the estimate. The effective sample size $\text{ESS} = (\sum_i w_i)^2 / \sum_i w_i^2$ measures this degeneracy; $\text{ESS} \ll N$ indicates severe weight concentration.

Sequential Monte Carlo (Section 18d.2) addresses weight degeneracy by resampling after each `observe` statement, maintaining a diverse population of particles. MCMC-based algorithms (SMC + MCMC kernels) further improve by combining resampling with local exploration.

0.121 18e. Type Theory, Dependent Types, and Program Verification

0.121.1 18e.1 Simply Typed Lambda Calculus

The **simply typed lambda calculus** ($\lambda^{\to}$; Church 1940) extends the untyped lambda calculus with a type discipline that prevents non-termination and ensures type safety. Types are built inductively from base types (e.g., ι for individuals, o for truth values) by the function type constructor: if σ and τ are types, then $\sigma \to \tau$ is the type of functions from σ to τ.

Typing judgements $\Gamma \vdash e : \tau$ (in context Γ, term e has type τ) are derived by the rules:

$$\frac{(x : \sigma) \in \Gamma}{\Gamma \vdash x : \sigma}\ \text{Var}, \qquad \frac{\Gamma, x : \sigma \vdash e : \tau}{\Gamma \vdash \lambda x : \sigma.\, e : \sigma \to \tau}\ \text{Abs}, \qquad \frac{\Gamma \vdash e_1 : \sigma \to \tau \quad \Gamma \vdash e_2 : \sigma}{\Gamma \vdash e_1 e_2 : \tau}\ \text{App}.$$

The **subject reduction theorem** (type safety): if $\Gamma \vdash e : \tau$ and $e \to e'$ (beta-reduction), then $\Gamma \vdash e' : \tau$. Types are preserved under computation. The **strong normalisation theorem**: every well-typed term in $\lambda^{\to}$ reduces to a unique normal form in finitely many steps. This means $\lambda^{\to}$ is a total language: all programs terminate.

The Curry-Howard correspondence (Section 22c) identifies $\lambda^{\to}$ with propositional logic: types are propositions, terms are proofs, beta-reduction is proof normalisation. Under this correspondence, strong normalisation corresponds to the cut-elimination theorem: every proof in propositional logic normalises to a cut-free form.

0.121.2 18e.2 Dependent Types and the Calculus of Constructions

Dependent types (Martin-Löf 1975; Coquand and Huet 1988) allow types to depend on values:

- **Dependent product** $\Pi_{x:A} B(x)$: the type of functions f such that $f(a) : B(a)$ for each $a : A$. If B does not depend on x, this reduces to the function type $A \to B$.
- **Dependent sum** $\Sigma_{x:A} B(x)$: the type of pairs (a, b) with $a : A$ and $b : B(a)$. If B is constant, this is the product type $A \times B$.

The **Calculus of Constructions** (CoC; Coquand and Huet 1988) is the type theory underlying the Coq proof assistant. It has sorts Prop (propositions) and Set (data types) and includes:

- $\Pi_{x:A} B$ for polymorphic functions and dependent function types.
- Inductive definitions (via fixpoint operators) for recursive types.
- Universe polymorphism for meta-level reasoning about types.

In CoC, proofs are programs and programs are proofs: to prove a theorem T in Coq, you write a program of type T, and the type-checker verifies correctness. This is the **propositions as types, proofs as programs** paradigm applied at industrial scale: the proof of the four-colour theorem in Coq (Gonthier 2005) is a verified program of $\sim$60,000 lines of Coq code.

0.121.3 18e.3 Probabilistic Types and the Probabilistic Calculus of Constructions

Extending dependent types to probabilistic computation gives a **probabilistic dependent type theory**: a type theory in which terms can be drawn from distributions, and types can depend on random values. The key challenge: if $a : A$ is a random term, then $B(a)$ is a random type, and the semantics of $\Pi_{x:A} B(x)$ (the type of probabilistic functions) requires careful measure-theoretic treatment.

The probabilistic Calculus of Constructions (Bagnall and Stewart 2019; Vákár et al., 2019) handles this via:

- Random variables as terms of a **probabilistic type** $\mathrm{Dist}(A)$: a distribution over terms of type A.
- Monadic bind `do` $x \leftarrow m;\ t$ (draw x from distribution m, then compute t) as the sequencing operator.
- Conditioning `observe` p (restrict the probability space to events where predicate p holds) as a measure-theoretic operation.

The semantic challenge is to define a category $\mathcal{C}$ in which probabilistic dependent types are interpretable. The correct setting is **quasi-Borel spaces** (Heunen et al., 2017): a category of measurable spaces with a richer structure that supports both random variables and function spaces. Quasi-Borel spaces form a **Cartesian closed category** (enabling function types) and support a strong probability monad (enabling monadic probabilistic computation) — properties that standard measurable spaces lack.

0.122 18c. Advanced Inference in Probabilistic Programs

0.122.1 18c.1 Inference Compilation

Inference compilation (Le et al., 2016; Ritchie et al., 2016) addresses the core bottleneck of general-purpose probabilistic programming: inference algorithms like MCMC and importance sampling must repeatedly execute the generative program, which is computationally expensive. Inference compilation replaces the inference algorithm with a trained **neural network** that predicts the posterior distribution given observations, *amortising* the inference cost.

The idea: given a probabilistic program $\mathcal{P}$ defining a generative model $p(x, z)$, compile an **inference network** $q_\phi(z|x)$ that approximates the posterior. Training proceeds by simulating from the prior (running $\mathcal{P}$ forward to generate (x, z) pairs) and maximising:

$$\mathcal{L}(\phi) = \mathbb{E}_{p(x,z)}[\log q_\phi(z \mid x)] \,.$$

This is the "sleep" phase of the Helmholtz machine (Hinton et al., 1995), applied to arbitrary probabilistic programs. At test time, inference is a single forward pass through q_ϕ: effectively instantaneous.

The architecture of q_ϕ must handle the variable-length execution traces of probabilistic programs. A natural choice is a **trace-aware recurrent network**: at each sampling site in the execution trace, the network conditions on all previously sampled values and produces a proposal distribution for the next sample. The LSTM or Transformer architecture encodes the execution history into a fixed-length state vector.

Inference compilation achieves the best of both worlds: the expressiveness of universal probabilistic programming (any computable generative model) with the speed of amortised variational inference (a single neural forward pass). The trade-off: the inference network is specific to a fixed observation space and may fail to generalise to observations far from the training distribution.

0.122.2 18c.2 Non-Parametric Bayesian Models as Probabilistic Programs

Non-parametric Bayesian models — models whose complexity grows with the data — can be expressed as probabilistic programs using infinite-dimensional random objects.

The **Dirichlet Process** (DP; Ferguson 1973) is a distribution over probability measures: $G \sim \text{DP}(\alpha, G_0)$ where $\alpha > 0$ is the concentration parameter and G_0 is the base measure. It can be defined via the **stick-breaking construction**:

$$v_k \sim \text{Beta}(1, \alpha), \qquad \pi_k = v_k \prod_{j<k}(1 - v_j), \qquad \theta_k \sim G_0, \qquad G = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k}.$$

As a probabilistic program:

```
def DP(alpha, G0):
    sticks = []
    atoms = []
    def sample():
        # Find first unbroken stick long enough
        remaining = 1.0
        for k in range(len(sticks)):
            if bernoulli(sticks[k] / remaining):
                return atoms[k]
            remaining -= sticks[k]
        # Create new stick
        v = beta(1, alpha)
        sticks.append(v * remaining)
        atoms.append(G0.sample())
        return atoms[-1]
    return sample
```

A **DP mixture model** assigns each observation x_i a cluster label $z_i \sim G$ (a DP sample) and observes $x_i | z_i \sim F(\theta_{z_i})$. The Chinese Restaurant Process (CRP) characterises the marginal distribution over cluster assignments: the nth customer sits at table k (existing cluster) with probability $n_k/(n-1+\alpha)$ and starts a new table with probability $\alpha/(n-1+\alpha)$. This yields a clustering that automatically infers the number of clusters from the data.

The **Indian Buffet Process** (IBP; Griffiths and Ghahramani 2011) is the analogous construction for binary feature matrices with an infinite number of potential features. The IBP defines a distribution over infinite binary matrices $Z \in \{0,1\}^{N \times \infty}$ where $Z_{nk} = 1$ iff object n possesses feature k. It is the de Finetti measure for exchangeable binary feature matrices with the "sparse infinity" property. As a probabilistic program, IBP inference underlies models of object segmentation, dictionary learning, and the representation of categorical knowledge with open-ended feature sets.

0.122.3 18c.3 Stan and Practical Probabilistic Programming

Stan (Carpenter et al., 2017) is the leading practical probabilistic programming system for Bayesian statistical modelling. Unlike Church or WebPPL, Stan is designed for *efficient* inference in statistical models rather than *universal* expressiveness. Its language supports continuous latent variables and uses the No-U-Turn Sampler (NUTS) as its primary inference algorithm, with variational inference (ADVI) and Laplace approximation as alternatives.

A Stan model is specified in blocks:

```
data { int<lower=0> N; vector[N] y; }
parameters { real mu; real<lower=0> sigma; }
model {
  mu ~ normal(0, 10);       // prior
  sigma ~ cauchy(0, 2.5);   // prior
  y ~ normal(mu, sigma);    // likelihood
}
```

Stan compiles the model's log-posterior $\log p(\theta|D)$ and its gradient $\nabla_\theta \log p(\theta|D)$ using **automatic differentiation** (reverse-mode AD), enabling NUTS to run at full efficiency. For a model with k parameters, AD computes the full gradient in $O(k)$ evaluations of the log-posterior — the same asymptotic cost as a single likelihood evaluation, regardless of k.

The **reparametrisation** of the prior is critical for NUTS performance. A *centred* parametrisation $\theta \sim \mathcal{N}(\mu, \sigma^2)$ creates a posterior geometry where θ and μ, σ^2 are strongly correlated; a *non-centred* parametrisation $\theta = \mu + \sigma\eta$, $\eta \sim \mathcal{N}(0, 1)$ decouples them, dramatically improving NUTS mixing. Stan's *diagnose* module detects problematic geometry (divergent transitions, poor $\hat{R}$) and suggests reparametrisations.

Section X

Section X — Cognitive Architectures: Mathematical Foundations

0.123 19. What Is a Cognitive Architecture?

A **cognitive architecture** is a fixed structural and procedural framework within which cognition takes place: the invariant part of the cognitive system, the part that does not change from task to task. Specific knowledge and strategies are encoded in the architecture's memories; the architecture itself specifies how memories are organised, how they interact, and how processing unfolds in time.

The major architectures — ACT-R, SOAR, and their descendants — are computational theories of the mind. They make precise, quantitative predictions about reaction times, error rates, and learning curves in psychological experiments. The goal of this Section is not to survey these architectures descriptively, but to extract their mathematical core: the structures and dynamics that underlie their empirical successes.

0.124 20. ACT-R: Activation, Memory, and Rational Analysis

0.124.1 20.1 Rational Analysis

ACT-R (Adaptive Control of Thought — Rational), developed by John Anderson, is grounded in the **rational analysis** methodology: rather than asking what mechanisms implement cognition, ask what cognitive behaviour would be optimal given the environment and the agent's goals, and then derive the mechanisms from that analysis.

The core of ACT-R's memory model is a Bayesian analysis of memory retrieval. The **activation** A_i of a memory chunk i at time t is:

$$A_i(t) = B_i + S_i + \epsilon_i,$$

where B_i is the **base-level activation**, S_i is the **associative activation** (spreading from the current context), and ϵ_i is a noise term. The base-level activation is the log-odds of needing chunk i:

$$B_i(t) = \ln\left(\sum_{j:t_{i,j}<t} (t - t_{i,j})^{-d}\right),$$

a sum over all past retrievals of chunk i at times $t_{i,j}$, weighted by a power-law decay with parameter d. This is the **base-level learning equation**: each retrieval strengthens the chunk's activation, but the strengthening decays as a power law of elapsed time. The power-law decay is motivated by the statistical structure of the environment: the probability of needing an item in the future is approximately proportional to how often and how recently it has been needed in the past, and the optimal predictor given this environmental structure is a power-law activation function.

The probability of retrieving chunk i when activation threshold τ is required:

$$P(\text{retrieve } i) = \frac{1}{1 + e^{(\tau - A_i)/s}},$$

a logistic function in the activation, where s is a noise scale parameter. The retrieval time is $RT = Fe^{-A_i}$, an exponential function of activation with parameter F. These equations link cognitive behaviour (response times) to latent memory quantities (activations) through closed-form expressions derivable from the Bayesian optimality principle.

0.124.2 20.2 Predictive Processing as Cognitive Architecture

The **predictive processing** (PP) framework, synthesising ideas from Helmholtz, Rao, Friston, and Clark, proposes that the fundamental computation of the brain is prediction and prediction-error minimisation across a hierarchy of representations. This is not merely a theory of perception: it is a candidate cognitive architecture, a claim about the universal computational motif of cognition.

Mathematically, predictive processing instantiates the hierarchical generative model of Section IV at the architectural level. A cortical hierarchy with L levels maintains, at level l, a set of **prediction units** $\mu^{(l)}$ and **prediction error units** $\varepsilon^{(l)}$:

$$\varepsilon^{(l)} = \Sigma^{(l)-1}(x^{(l)} - g^{(l)}(\mu^{(l+1)})),$$

$$\dot{\mu}^{(l)} = \frac{\partial \mathcal{F}}{\partial \mu^{(l)}} = -(J_{g^{(l)}})^{\top}\varepsilon^{(l)} + \varepsilon^{(l-1)},$$

where $x^{(l)}$ is the representation at level l, $g^{(l)}$ is the generative function from level $l+1$ to level l (implemented by cortical feedback connections), $\Sigma^{(l)}$ is the predicted precision (inverse variance) at level l, and $J_{g^{(l)}}$ is the Jacobian. The dynamics are gradient descent on the variational free energy, unfolded through the cortical hierarchy. This connects the abstract mathematics of Section IV to the empirical anatomy and physiology of the cortex.

0.125 20b. SOAR: Productions, Problem Spaces, and Chunking

0.125.1 20b.1 The SOAR Architecture

SOAR (State, Operator And Result; Laird, Newell, Rosenbloom 1987) is a symbolic cognitive architecture built around **production rules** and **problem spaces**. Its fundamental cycle is:

1. **Elaboration:** Fire all applicable production rules in parallel to augment working memory with new elements.
2. **Decision:** Select an operator from the proposed operators using a preference system.
3. **Application:** Apply the selected operator to transform the current state.
4. **Impasse resolution:** If no operator can be selected, create a subgoal to resolve the impasse by further search.

Working memory in SOAR consists of **objects** with **attribute-value** pairs. Production rules match patterns of working memory objects and fire to add, remove, or modify elements. The **preference system** uses a lattice of preferences (better, worse, indifferent, best, worst, required, prohibited) to select operators; if preferences conflict or are incomplete, an **impasse** occurs and a subproblem is created.

Chunking is SOAR's learning mechanism: when an impasse is resolved, a new production rule ("chunk") is created that directly matches the conditions that led to the impasse and produces the result that resolved it, preventing future impasses of the same type. Chunking implements a form of **explanation-based learning**: the learned chunk compresses the search that resolved the impasse into a direct look-up.

Mathematically, chunking can be analysed using the theory of **lazy evaluation** and **memoisation**: the first time a computation is performed (via search), its result is stored as a production rule; subsequent

identical inputs are resolved by pattern matching in $O(\text{rule length})$ rather than $O(\text{search depth})$.

0.125.2 20b.2 Mathematical Cognitive Architectures: Timing and Reaction Times

Cognitive architectures make quantitative predictions about reaction times that can be tested experimentally. For ACT-R and SOAR, timing models specify the duration of each processing stage:

Fitts' Law predicts movement time as a logarithmic function of target distance and size:

$$\text{MT} = a + b \log_2\left(\frac{2D}{W}\right) = a + b \cdot \text{ID},$$

where D is the distance to the target, W is the target width, and $\text{ID} = \log_2(2D/W)$ is the **index of difficulty**. The logarithmic form reflects the optimal strategy for a noisy motor system: each corrective sub-movement halves the remaining distance, generating a logarithmic number of corrections for a given accuracy criterion.

The mathematical derivation from signal detection theory: if motor noise is proportional to movement amplitude (signal-dependent noise: $\sigma \propto A$), then the optimal strategy produces movements that halve the remaining error at each step. After k sub-movements, the residual error is $\propto (1/2)^k D$, which reaches threshold $W/2$ at $k = \log_2(2D/W)$ steps. Each step takes time $\propto A_k \propto D(1/2)^k$, and summing gives $\text{MT} \propto D \sum_k (1/2)^k = O(\log(D/W))$.

Hick-Hyman Law predicts choice reaction time as a linear function of the log-number of equally likely alternatives:

$$\text{RT} = a + b \log_2(n),$$

where n is the number of equally likely choices. The information-theoretic interpretation: the decision requires $\log_2(n)$ bits of information; if each bit is processed in constant time b, total RT is linear in information content. For unequal probabilities: $\text{RT} = a + bH(X)$ where $H(X) = -\sum_i p_i \log_2 p_i$ is the stimulus entropy. The law fails when n is very large or when stimuli are meaningfully grouped; corrections involving sequential sampling models (drift-diffusion) provide more accurate predictions.

The Drift-Diffusion Model (DDM) provides a continuous-time, mechanistic account of two-alternative forced-choice tasks. Evidence for alternative A versus B accumulates as a stochastic process $x(t)$ with:

$$\frac{dx}{dt} = \nu + \varepsilon(t), \qquad \varepsilon(t) \sim \mathcal{N}(0, \sigma^2),$$

where ν is the **drift rate** (the rate of evidence accumulation for the correct alternative) and σ^2 is the diffusion coefficient (moment-to-moment noise). A decision is made when $x(t)$ reaches one of two absorbing boundaries at $\pm a$ (threshold). The first-passage time distribution to boundary $+a$ has:

$$\mathbb{E}[\text{RT} \mid \text{correct}] = \frac{a}{2\nu} \tanh\!\left(\frac{\nu a}{2\sigma^2}\right), \qquad P(\text{correct}) = \frac{1}{1 + e^{-2\nu a/\sigma^2}}.$$

The speed-accuracy tradeoff is parametrised by the boundary a: wider boundaries increase accuracy but slow responses. The DDM is the optimal algorithm for a binary decision under the Bayes-optimal criterion of maximising expected reward rate (Bogacz et al., 2006): the integral of the posterior log-odds ratio is a sufficient statistic, and the optimal policy is to wait until it reaches a threshold. This provides a normative justification for the neural circuitry of decision-making (leaky competing accumulators in the

basal ganglia and prefrontal cortex).

0.126 20c. Global Workspace Theory: A Mathematical Formalisation

0.126.1 20c.1 The Workspace Architecture

Global Workspace Theory (GWT; Baars 1988; Dehaene and Changeux 2004) proposes that consciousness arises from the broadcast of selected information from a centralised "workspace" to specialised processing modules. The mathematical structure of this theory can be formalised as a **bottleneck network architecture**.

The global workspace is a high-capacity, limited-bandwidth communication channel that receives inputs from specialised modules (perception, memory, attention, affect) and broadcasts the selected content to all modules simultaneously. The selection process is competitive: multiple modules compete to write their contents to the workspace, and the "winning" content is broadcast globally.

Formal model. Let $\mathcal{M} = \{M_1, \ldots, M_K\}$ be the set of specialised modules. Each module M_k maintains an internal state $s_k \in \mathcal{S}_k$ and can attempt to broadcast a message $m_k \in \mathcal{C}$ (the workspace vocabulary). The workspace selects the message with the highest "ignition strength":

$$m^* = \underset{k \in \{1,\ldots,K\}}{\arg\max}\ \gamma_k(s_k, c_k),$$

where $\gamma_k(s_k, c_k)$ is the ignition strength of module k's message given its internal state s_k and contextual input c_k (from other modules). The selected message m^* is broadcast to all modules, updating their states:

$$s_k^{(t+1)} = f_k(s_k^{(t)}, m^*),$$

with module-specific update functions f_k.

The **ignition** process — the sudden, global activation of the workspace — is modelled as a first-order phase transition in the network's dynamics. Below a threshold of bottom-up activation, local processing occurs without global broadcast; above threshold, the workspace ignites and the content becomes globally available. Dehaene et al. (2003) modelled this as a **supercritical bifurcation** in a network of excitatory-inhibitory neuronal populations.

0.126.2 20c.2 Mathematical Analysis of Workspace Dynamics

The workspace's capacity constraint is central to GWT's predictions about conscious access. If the workspace has capacity C (bits), then at most C bits of information can be globally broadcast at any moment. The **information bottleneck** (IB; Tishby et al., 1999) framework provides a variational characterisation:

The IB objective for a bottleneck variable T (the workspace representation) that captures relevant information Y from input X:

$$\mathcal{L}_{\text{IB}} = \text{I}(T; Y) - \beta\, \text{I}(T; X),$$

where $\beta > 0$ controls the tradeoff between relevance and compression. The optimal T minimises $-\mathcal{L}_{\text{IB}}$: it compresses X to T (low $\text{I}(T; X)$) while retaining all information about Y (high $\text{I}(T; Y)$).

The IB curve I_Y vs. I_X traces the Pareto frontier of achievable compressions. Points on the curve are obtained by varying β; the optimal representation for a given compression rate $I_X = C$ is the point on the IB curve with $\mathrm{I}(T;X) = C$. For Gaussian sources, the IB curve is computed analytically using the water-filling algorithm; for general sources, the Blahut-Arimoto algorithm iterates:

$$q_{t+1}(t|x) \propto q_t(t)\exp(\beta \cdot \log q_t(y|t)/p(y|x)),$$

$$q_{t+1}(t) = \sum_x p(x) q_{t+1}(t|x).$$

GWT's prediction is that consciousness corresponds to the global broadcast of information at a specific compression level $C^* = C_{\text{workspace}}$. Subliminal (unconscious) processing occupies points with $\mathrm{I}(T;X) < C^*$ (too compressed to broadcast), while conscious processing requires $\mathrm{I}(T;X) \geq C^*$.

0.126.3 20c.3 Predictive Processing Architecture: Neural Implementation

The full hierarchical predictive processing architecture (Section 20.2) can be analysed in terms of its convergence properties and computational complexity.

For an L-level hierarchy with state variables $\mu^{(l)} \in \mathbb{R}^{d_l}$ and precision matrices $\Pi^{(l)} \in \mathbb{R}^{d_l \times d_l}$, the free-energy gradient dynamics form a coupled system of $\sum_l d_l$ differential equations. The system has a unique fixed point when the generative functions $g^{(l)}$ are smooth and the precisions are finite; the fixed point is the posterior mode of the full hierarchical model.

Convergence rate. The linearised dynamics around the fixed point have Jacobian:

$$J = -\begin{pmatrix} \Pi^{(1)} + (J_{g^{(1)}})^\top \Pi^{(2)} J_{g^{(1)}} & -(J_{g^{(1)}})^\top \Pi^{(2)} \\ -\Pi^{(2)} J_{g^{(1)}} & \Pi^{(2)} + \cdots \end{pmatrix},$$

which is negative definite when the precisions are positive definite and the generative functions have bounded Jacobians. The convergence rate is governed by the smallest eigenvalue of $-J$: faster convergence for higher precisions, slower for more complex generative functions.

The temporal depth of inference is determined by how quickly the hierarchy equilibrates. For a hierarchy with L levels and similar dynamics at each level, the settling time scales as $O(L^2)$ (propagation from the bottom to the top takes $O(L)$ time steps, and corrections propagate back down in another $O(L)$ steps). This predicts that deeper cognitive processing (involving higher cortical areas) takes longer — a prediction consistent with the known latencies of neural responses in different cortical areas.

0.127 20d. Mathematical Models of Reaction Time and Decision Making

0.127.1 20d.1 Sequential Probability Ratio Tests

The **Sequential Probability Ratio Test** (SPRT; Wald 1947) is the optimal statistical procedure for deciding between two hypotheses $H_0 : \theta = \theta_0$ and $H_1 : \theta = \theta_1$ from sequentially observed data, in the sense of minimising expected sample size subject to error-rate constraints.

At each step n, compute the log-likelihood ratio:

$$\Lambda_n = \sum_{i=1}^{n} \log \frac{p(x_i|\theta_1)}{p(x_i|\theta_0)}.$$

Stop and decide H_1 when $\Lambda_n \geq B = \log \frac{1-\beta}{\alpha}$; decide H_0 when $\Lambda_n \leq A = \log \frac{\beta}{1-\alpha}$; continue sampling otherwise. Here α and β are the type-I and type-II error rates.

Theorem 0.127.1 (Wald-Wolfowitz 1948)**.** Among all sequential tests with error rates $\leq \alpha$ (type I) and $\leq \beta$ (type II), the SPRT minimises the expected sample size under both H_0 and H_1.

The Drift-Diffusion Model (DDM) is the continuous-time limit of the SPRT: replace the discrete sum Λ_n with a Brownian motion with drift ν (the evidence accumulation rate) and diffusion σ^2. The decision boundaries A and B become the absorbing boundaries $\pm a$ of the DDM. The DDM is thus the optimal Bayesian decision rule for a binary choice problem with normally distributed incremental evidence. The neural circuits implementing the DDM (accumulate-to-threshold in the LIP and basal ganglia) are implementing an optimal statistical procedure.

0.127.2 20d.2 Race Models and Independence

For choices among $K > 2$ alternatives, the **race model** (horse race model) posits that each alternative k has its own accumulator $x_k(t)$ with drift ν_k and diffusion σ_k^2:

$$\frac{dx_k}{dt} = \nu_k + \sigma_k \varepsilon_k(t), \qquad \varepsilon_k(t) \sim \mathcal{N}(0,1) \text{ i.i.d.}$$

The decision is made by the first accumulator to reach threshold a: response is $k^* = \arg\min_k T_k$ where T_k is the first passage time of $x_k(t)$ to a.

For the race model, the **survivor interaction contrast** (SIC) provides a non-parametric test of whether the accumulators are statistically independent. Define $S_k(t) = P(\text{RT} > t | \text{correct response is } k)$. For independent accumulators:

$$\text{SIC}(t) = S_{HH}(t) - S_{HL}(t) - S_{LH}(t) + S_{LL}(t) \equiv 0,$$

where HH, HL, etc. denote conditions with high/low salience on dimension 1 and 2. Non-zero SIC indicates statistical interaction between accumulators, violating the race model.

The race model provides a geometric interpretation of choice probabilities. The probability of choosing alternative k is:

$$P(k \text{ wins}) = P(T_k < \min_{j \neq k} T_j),$$

which for independent Brownian motions (no inter-accumulator correlations) can be computed via the multivariate first-passage time distribution. For equal drift rates and thresholds, $P(k \text{ wins}) = 1/K$ (uniform distribution); differences in drift rates (salience) bias the probabilities in proportion to $\exp(\nu_k a / \sigma_k^2)$.

Section *XI*

Section XI — Language, Formalism, and Categorical Cognition

0.128 21. The Compositionality Problem in Cognition

Human cognition is **systematically compositional**: we understand sentences we have never heard before because we understand their parts and the rules for combining them. We represent novel objects by composing familiar conceptual parts. We solve new problems by decomposing them into known subproblems. This **systematicity**, as Fodor and Pylyshyn argued, is a structural feature of cognition that requires explanation.

This Section develops two mathematical frameworks for compositionality in cognition. The first is Montague's compositional semantics: a rigorous theory of how the meaning of a sentence is a function of the meanings of its parts and their mode of combination, expressed in the lambda calculus. The second is the categorical framework of DisCoCat and related approaches: a generalisation using monoidal categories and functors to give a unified algebraic treatment of syntax, semantics, and their relationship.

0.129 22. Montague Grammar: Intensional Logic and Compositional Semantics

0.129.1 22.1 The Typed Lambda Calculus

Montague's approach rests on a typed formal language: **Intensional Logic** (IL). Every expression has a type, and types are built from two base types: e (entities, individuals in the domain) and t (truth values). Complex types are constructed by: (i) if α and β are types, then $\langle \alpha, \beta \rangle$ is the type of a function from α to β; (ii) if α is a type, then $\langle s, \alpha \rangle$ is the intension type (functions from possible worlds to α-type objects).

Common noun phrases like "run" have type $\langle e, t \rangle$: a function from individuals to truth values (the characteristic function of the set of running entities). A transitive verb like "love" has type $\langle e, \langle e, t \rangle \rangle$: a function from the object to a property (the set of subjects that love that object). A quantificational NP like "every dog" has type $\langle \langle e, t \rangle, t \rangle$: a function from a property to a truth value (true iff every dog has that property, i.e., a generalised quantifier).

The **lambda abstraction** $\lambda x_\alpha . M_\beta$ has type $\langle \alpha, \beta \rangle$ and denotes the function mapping each α-type object x to the β-type object denoted by M under the assignment x. The **function application** $M_{\langle \alpha,\beta \rangle} N_\alpha$ has type β and denotes the result of applying the function denoted by M to the argument denoted by N. The **beta-reduction rule**: $(\lambda x.M)N$ reduces to $M[x := N]$, the substitution of N for x in M.

0.129.2 22.2 Compositional Semantics: The Homomorphism Principle

The central principle of Montague semantics is that **syntax and semantics are homomorphic**: there is a structure-preserving map from syntactic derivations to semantic values. If α is a syntactic constituent with parts $\beta_1, \ldots, \beta_n$ combined by rule r, then the semantic value $[\![\alpha]\!]$ is a function of $[\![\beta_1]\!], \ldots, [\![\beta_n]\!]$:

$$[\![r(\beta_1, \ldots, \beta_n)]\!] = F_r([\![\beta_1]\!], \ldots, [\![\beta_n]\!]).$$

Each syntactic rule r has a corresponding semantic operation F_r. The most common is function application: if $\alpha = \beta\gamma$ and β has type $\langle \alpha', \beta' \rangle$ while γ has type α', then $[\![\alpha]\!] = [\![\beta]\!]([\![\gamma]\!])$. This reduces semantic composition to functional application.

Example. Consider "every dog runs." The NP "every dog" denotes a generalised quantifier $\lambda P. \forall x(\text{Dog}(x) \Rightarrow$

$P(x))$ of type $\langle\langle e,t\rangle,t\rangle$. The VP "runs" denotes the property $\lambda x.\,\text{Run}(x)$ of type $\langle e,t\rangle$. The sentence denotation is function application:

$$[\![\text{every dog runs}]\!] = [\![\text{every dog}]\!]([\![\text{runs}]\!]) = (\lambda P.\,\forall x(\text{Dog}(x) \Rightarrow P(x)))(\lambda x.\,\text{Run}(x)).$$

Beta-reducing: $\forall x(\text{Dog}(x) \Rightarrow \text{Run}(x))$. This is the correct truth condition: the sentence is true iff every entity that is a dog is also a runner.

0.130 23. Categorical Compositional Distributional Semantics

0.130.1 23.1 Monoidal Categories

The categorical approach to compositional semantics, developed by Coecke, Sadrzadeh, and Clark (DisCoCat), embeds both syntax and semantics in the language of **monoidal categories**.

Definition 0.130.1 (Monoidal Category)**.** A monoidal category $(\mathcal{C}, \otimes, I)$ consists of a category $\mathcal{C}$ with a bifunctor $\otimes : \mathcal{C} \times \mathcal{C} \to \mathcal{C}$ (the tensor product) and a unit object I, together with natural isomorphisms: $\alpha_{A,B,C} : (A \otimes B) \otimes C \cong A \otimes (B \otimes C)$ (associativity), $\lambda_A : I \otimes A \cong A$ and $\rho_A : A \otimes I \cong A$ (unit laws), satisfying coherence conditions (MacLane's pentagon and triangle equations).

A **compact closed category** is a monoidal category in which every object A has a dual A^* with unit $\eta_A : I \to A^* \otimes A$ and counit $\epsilon_A : A \otimes A^* \to I$ satisfying the snake equations:

$$(\epsilon_A \otimes 1_A) \circ (1_A \otimes \eta_A) = 1_A, \quad (1_{A^*} \otimes \epsilon_A) \circ (\eta_A \otimes 1_{A^*}) = 1_{A^*}.$$

Compact closed categories have a complete **string diagram** calculus: morphisms are represented as boxes connected by strings, with the tensor product as juxtaposition and composition as connection. The snake equations correspond to the ability to "straighten" loops in diagrams.

0.130.2 23.2 DisCoCat: Syntax Meets Semantics Categorically

In DisCoCat, the syntactic category is the **pregroup grammar** (a compact closed category of grammatical types) and the semantic category is **FdVect** (the compact closed category of finite-dimensional vector spaces and linear maps). The functor $F : \text{Syn} \to \textbf{FdVect}$ maps:

- each basic type a to a vector space $F(a)$: noun types n to a semantic space N, sentence types s to a sentence space S;
- each grammatical reduction (type contraction via η or ϵ) to a linear map (via $F(\eta)$ and $F(\epsilon)$);
- each word to a vector (for nouns) or tensor (for verbs, quantifiers): a transitive verb of type $n^r s n^l$ maps to a tensor in $N \otimes S \otimes N$.

The functor F is the formal realisation of the homomorphism principle in categorical language: it maps syntactic composition to semantic tensor contraction. The sentence meaning is computed by applying F to the syntactic derivation diagram. For "dogs run", the noun "dogs" maps to a vector $\mathbf{n} \in N$, the verb "run" to a matrix $W_{\text{run}} \in N \otimes S$, and the sentence meaning to the vector $(\mathbf{n}^\top \otimes 1_S)W_{\text{run}} \in S$ — a projection of the verb matrix onto the subject direction.

This framework has a striking property: it makes the semantics of language **compositional by construction**. The functor is a theorem, not a heuristic. And because vector spaces are concrete mathematical objects with geometric structure, the sentence meanings are vectors in a high-dimensional space, amenable to similarity computation, clustering, and neural implementation.

0.131 20g. Mathematical Models of Attention

0.131.1 20g.1 Signal Detection Theory for Attention

Signal Detection Theory (SDT; Green and Swets 1966) provides the foundational mathematical framework for attention and perception. An observer detects a signal in noise; the decision is a function of the sensory evidence x compared to a criterion c:

- **Hit**: signal present (H_1) and $x \geq c$ — respond "yes."
- **Miss**: signal present (H_1) and $x < c$ — respond "no."
- **False alarm**: signal absent (H_0) and $x \geq c$ — respond "yes."
- **Correct rejection**: signal absent (H_0) and $x < c$ — respond "no."

For Gaussian distributions $p(x|H_0) = \mathcal{N}(0,1)$ and $p(x|H_1) = \mathcal{N}(d',1)$:

$$P(\text{hit}) = 1 - \Phi(c - d'), \qquad P(\text{FA}) = 1 - \Phi(c).$$

The discriminability $d' = \mu_1 - \mu_0$ (in units of the common $\sigma = 1$) is the fundamental measure of perceptual sensitivity, invariant to the observer's criterion c. The criterion c reflects motivational factors (reward for hits, cost of false alarms) and is set at the Bayesian optimal point $c^* = \frac{1}{d'} \log \frac{P(H_0)}{P(H_1)} + \frac{d'}{2}$ when maximising expected reward.

Attention and d'. Attention improves d' (perceptual sensitivity) rather than shifting the criterion c. This is the SDT operationalisation of the "spotlight" metaphor: attention enhances the signal-to-noise ratio for attended stimuli, increasing the Gaussian separation d'. Mathematically, attention multiplies the signal gain $\sigma_1^2 \to \sigma_1^2/\gamma$ (reduces variance in the attended channel), giving $d' \propto \sqrt{\gamma}$.

0.131.2 20g.2 The Attentional Blink: Temporal Bottleneck Models

The **attentional blink** (AB; Raymond et al., 1992) is the failure to detect the second of two targets (T2) when it appears 200–500 ms after the first (T1) in a rapid serial visual presentation (RSVP) stream. The AB demonstrates a temporal bottleneck in attention: processing T1 consumes cognitive resources for $\sim$500 ms, during which T2 representation is lost if it falls outside the attentional window.

The Boost-and-Bounce model (Olivers and Meeter 2008) formalises this as a neural excitation-inhibition system. After T1, a brief excitatory boost enables T1 identification; if T2 arrives during the boost, it is consolidated. However, the boost is followed by an inhibitory rebound that suppresses inputs for $\sim$300 ms: T2 arriving during the inhibitory rebound is suppressed.

The mathematical dynamics: let $s(t)$ be the attention signal (excitatory = positive, inhibitory = negative),

$i(t)$ be the inhibitory state. The boost-and-bounce dynamics:

$$\tau_s \dot{s}(t) = -s(t) + I(t) - \beta i(t), \qquad \tau_i \dot{i}(t) = -i(t) + \max(s(t), 0),$$

where $I(t)$ is the stimulus input (impulse at T1 time), β is the inhibitory gain, and $\tau_s < \tau_i$ (fast excitation, slow inhibition). The model predicts:

- T2 identification probability $P(\text{T2}) = \Phi(s(t_{\text{T2}}) - \theta)$ where θ is a threshold.
- The AB depth (minimum $P(\text{T2})$) depends on β and τ_i.
- The lag-1 sparing (high $P(\text{T2})$ at very short T1–T2 lags) occurs because T2 arrives during the boost.

0.131.3 20g.3 Spatial Attention: the Biased Competition Model

The **biased competition model** (Desimone and Duncan 1995; Reynolds and Chelazzi 2004) proposes that neural representations of stimuli compete for limited processing resources, and top-down attention provides a bias signal that enhances representations of attended objects.

Mathematically, for two competing stimuli s_1 and s_2 represented by neural populations with activities r_1 and r_2:

$$\tau \dot{r}_1 = -r_1 + \frac{\alpha_1 s_1}{1 + \gamma(r_1 + r_2)} + b_1,$$

$$\tau \dot{r}_2 = -r_2 + \frac{\alpha_2 s_2}{1 + \gamma(r_1 + r_2)} + b_2,$$

where α_i are sensory gain factors, γ is the competition term (mutual suppression), and b_i are attentional bias signals. The competition term $\gamma(r_1 + r_2)$ normalises the total activity: as one representation grows, it suppresses the other.

At steady state, the competition term determines the winner. Without bias ($b_1 = b_2 = 0$), the stronger stimulus ($\alpha_1 s_1 > \alpha_2 s_2$) wins. With attention bias ($b_1 > 0$), the attended stimulus wins even if it is weaker. The biased competition model predicts:

- **Neural suppression**: activity evoked by an unattended stimulus in the presence of an attended stimulus is lower than when the unattended stimulus appears alone (response suppression).
- **Contrast enhancement**: the attended stimulus is represented with higher contrast than when unattended.
- **Attentional cueing**: top-down bias signals from prefrontal cortex (the source of b_i) are necessary and sufficient to resolve competition between equally salient stimuli.

0.132 20e. Neural Networks as Cognitive Architectures

0.132.1 20e.1 The Universal Approximation Theorem

The **Universal Approximation Theorem** (Cybenko 1989; Hornik 1991) establishes that shallow neural networks are dense in function spaces:

Theorem 0.132.1 (Universal Approximation; Cybenko 1989)**.** Let $\sigma : \mathbb{R} \to \mathbb{R}$ be a continuous sigmoidal function ($\lim_{t\to+\infty}\sigma(t) = 1$, $\lim_{t\to-\infty}\sigma(t) = 0$). For any $f \in C([0,1]^d)$ (continuous function on the d-dimensional unit cube) and $\epsilon > 0$, there exist $N \in \mathbb{N}$, weights $w_j \in \mathbb{R}^d$, biases $b_j \in \mathbb{R}$, and output weights $c_j \in \mathbb{R}$ such that:

$$\sup_{x\in[0,1]^d} \left| f(x) - \sum_{j=1}^{N} c_j \sigma(w_j^\top x + b_j) \right| < \epsilon.$$

The theorem guarantees existence but not constructibility: it does not say how to find the approximating network. It is also a qualitative result — any target function can be approximated, but the required network size N may grow exponentially in dimension d (the **curse of dimensionality**).

The **depth separation theorem** (Eldan and Shamir 2016; Telgarsky 2016) shows that deep networks can represent functions exponentially more efficiently than shallow networks: there exist functions expressible by a depth-3 network of polynomial size that require exponential size for depth-2 networks. The mathematical reason: depth allows the repeated composition of functions, enabling the representation of hierarchical structure. A function of the form $f(x) = g(h_1(x), h_2(x), h_3(x))$ can be represented by a depth-3 network with $O(N)$ neurons if each of g, h_1, h_2, h_3 can be represented by $O(N/3)$ neurons; a depth-2 representation would need $O(N \cdot M)$ neurons where M is the complexity of the interactions between components.

For cognitive architecture, the depth separation result supports the view that the brain's hierarchical organisation is computationally advantageous: deeper cortical hierarchies can represent more complex transformations between sensory inputs and cognitive outputs with fewer total neurons.

0.132.2 20e.2 Transformer Architecture as Approximate Bayesian Inference

The **Transformer** architecture (Vaswani et al., 2017) has become the dominant architecture for language modelling, reasoning, and multi-modal AI. Its core operation, **scaled dot-product attention**, has a surprising connection to Bayesian inference.

The attention mechanism computes, for query Q, key K, and value V matrices:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right) V.$$

Each row of the output is a weighted sum of the rows of V, with weights given by the softmax of the inner products between the query and the keys. This is a **soft nearest-neighbour retrieval**: find the key most similar to the query (highest inner product), and return the corresponding value, with a smooth interpolation over all keys.

The Bayesian interpretation (Ramsauer et al., 2020): attention is a mean-field update for a hierarchical Hopfield network. The energy function $E(x, \xi^1, \ldots, \xi^N) = -x^\top \sum_i \xi^i \sigma(\xi^{i\top} x)$ (where ξ^i are stored patterns and x is the query) has fixed points at positions that minimise E. The update rule $x \leftarrow \sum_i p_i \xi^i$ where $p_i = \text{softmax}(\xi^{i\top} x)$ is precisely the attention mechanism. Memory retrieval in a Transformer is a single step of Hopfield network energy minimisation.

The **in-context learning** ability of large Transformers (learning from examples provided in the input context, without gradient updates) has been analysed as implicit Bayesian inference: the Transformer

effectively performs posterior predictive inference for a linear model, with the in-context examples serving as the training data for an online Bayesian update. Formally, for a linear regression model with Gaussian prior, the posterior predictive mean equals the ridge regression estimate, which can be computed by a single attention layer (Akyürek et al., 2022).

0.133 20m. Mathematical Models of Cognitive Development

0.133.1 20m.1 Stochastic Grammar Induction

Grammar induction — learning the syntactic rules of a language from example sentences — is one of the central problems in cognitive development. The Bayesian approach (Perfors et al., 2011; Huth et al., 2016) models the child as performing Bayesian inference over a space of probabilistic context-free grammars (PCFGs):

$$p(G|D) \propto p(D|G)p(G) = \prod_{s} p(s|G) \cdot p(G),$$

where D is the observed speech corpus, s ranges over sentences in D, and G is a PCFG.

The prior $p(G)$ encodes preferences for simpler grammars (fewer rules, fewer nonterminals) via a Dirichlet process prior over grammar rules: $p(\text{rules for nonterminal } X) = \text{Dir}(\alpha)$ where α is the concentration. The likelihood $p(s|G)$ is the inside probability of sentence s under grammar G, computed by the CYK algorithm in $O(n^3|G|)$ time.

The poverty of the stimulus argument revisited. Chomsky (1965) argued that the child's grammatical knowledge cannot be induced from the impoverished input (the poverty of the stimulus). The Bayesian counter-argument (Perfors et al., 2011): if the prior over grammars is appropriately structured (hierarchical, compositional), the posterior correctly identifies the underlying grammar even from finite input. The prior does real work: it's not that the child has no inductive bias, but that the inductive bias is the prior, and a well-chosen prior enables rapid grammar learning.

0.133.2 20m.2 Theory Learning and Conceptual Development

Naïve theories are intuitive, coherent systems of knowledge about domains (naïve physics, naïve biology, naïve psychology) that young children develop before formal education. The Bayesian program for theory learning (Tenenbaum et al., 2011) models theory learning as Bayesian inference over a space of causal frameworks:

$$p(\text{theory}|D) \propto p(D|\text{theory})p(\text{theory}).$$

A **theory** is a structured set of causal principles (e.g., "objects move continuously," "unsupported objects fall," "animate objects have goals"). The likelihood $p(D|\text{theory})$ measures how well the theory predicts the observed data; the prior favours simpler, more compact theories.

Theory changes (conceptual revolutions in development, like the shift from impetus physics to Newtonian physics) are modelled as large jumps in the posterior, triggered by sufficient evidence against the current theory. This is Bayesian belief revision at the level of entire conceptual frameworks: not just updating parameter values, but switching between qualitatively different generative models. The developmental trajectory is the Bayesian posterior over theories as evidence accumulates with age.

0.134 20l. Working Memory: Capacity and Forgetting

0.134.1 20l.1 The Magic Number Seven

Miller's law (Miller 1956): the capacity of short-term (working) memory is approximately 7 ± 2 chunks. A **chunk** is a familiar unit that can be represented as a single entity; chunking compresses the representation, effectively increasing the number of items that can be held.

The information-theoretic formulation: the capacity of working memory is approximately $C \approx 7$ **chunks**, where each chunk carries H_{chunk} bits of information. The total information capacity is $C_{\text{info}} = 7 \times H_{\text{chunk}}$ bits. For binary items (each chunk is one bit), $C_{\text{info}} = 7$ bits; for larger chunks (e.g., words, each carrying ~ 10 bits of information), $C_{\text{info}} \approx 70$ bits.

More recent evidence (Cowan 2001) suggests the capacity is closer to 4 ± 1 chunks (not 7), and that Miller's estimate was inflated by rehearsal strategies. The **focus of attention** model (Oberauer 2002) distinguishes: (1) the focus of attention (1 item, fully activated), (2) the region of direct access (3–4 items, activated but not in focus), and (3) activated long-term memory (many items, above-threshold activation). These correspond to different activation levels in ACT-R (Section 20f): items in the focus of attention have activation $\gg \tau$, items in the region of direct access have activation $> \tau$, and items in activated LTM have activation near τ.

0.134.2 20l.2 Mathematical Models of Forgetting

Ebbinghaus' forgetting curve (1885): memory retention $R(t) = e^{-t/S}$ decays exponentially with time t, with time constant S (the "strength" of the memory). Modern versions (Wickelgren 1974; Anderson and Lebière 1998) use power-law forgetting: $R(t) = (1+bt)^{-d}$ where b and d are parameters. The power law fits memory retention data better than exponential decay, especially over very long timescales.

The ACT-R base-level learning equation (Section 20f) gives $B_i(t) \approx \log(n_i/(t - t_{i,1})^d)$ for memory i with n_i uses over interval $[t_{i,1}, t]$. The forgetting is power-law in recency, consistent with the empirical forgetting curve.

Spacing effect. Distributing practice trials over time (spaced practice) produces better retention than massing practice (massed practice), a robust empirical finding known as the spacing effect. The ACT-R explanation: spaced practice produces a higher base-level activation at test time because the sum $\sum_j t_j^{-d}$ is higher for temporally distributed presentations (the recency contributions are more equal) than for massed presentations (dominated by the single most recent presentation). The optimal spacing for n practice trials over a retention interval $[0, T]$ maximises $B(T) = \log(\sum_{j=1}^{n}(T - t_j)^{-d})$ over the schedule $t_1 < t_2 < \cdots < t_n$.

0.135 20k. Embodied Simulation and Motor Cognition

0.135.1 20k.1 Mirror Neurons and Action Understanding

Mirror neurons (Rizzolatti et al., 1996; di Pellegrino et al., 1992) are neurons in the premotor cortex (F5 in macaques; IFG/PMv in humans) that fire both when an agent performs an action and when it observes another agent performing the same action. The **direct matching hypothesis** proposes that action understanding is achieved by simulating the observed action in the observer's motor system.

The Active Inference formalisation (Kilner et al., 2007): the observation of another agent's movement is explained by the same hierarchical generative model used for self-generated movement. The top-level causes are the agent's goals (reaching for an object); the middle level are the kinematics (arm trajectory); the bottom level are the motor commands. When observing another agent, the observer's predictive coding hierarchy generates predictions about the observed movements; mirror neuron activity reflects the precision-weighted prediction errors at the level of motor commands.

Forward model and emulation. The brain's **forward model** $f : (\text{current state}, \text{action}) \to \text{predicted next state}$ is used both for self-generated movement (predicting the consequences of one's own actions) and for action observation (predicting the consequences of another agent's actions, assuming they have the same motor system). The forward model is shared for self and other: this is the computational basis of the **simulation theory of mind**.

0.135.2 20k.2 Corollary Discharge and Sensory Attenuation

When the motor system generates a movement command, it simultaneously sends a **corollary discharge** (efference copy) to sensory systems, predicting the expected sensory consequences of the movement. This prediction is used to **attenuate** (suppress) the sensory feedback generated by self-produced movements.

Formally, the corollary discharge generates a sensory prediction $\hat{\mathbf{s}}_{\text{motor}} = g_{\text{fwd}}(\mathbf{u})$ (the expected sensory state given motor command $\mathbf{u}$). The precision of this prediction Π_{motor} determines how much the sensory input is attenuated: high precision $\to$ strong suppression; low precision $\to$ weak suppression. The resulting sensory prediction error $\tilde{\mathbf{s}} - \hat{\mathbf{s}}_{\text{motor}}$ is reduced by self-generated movements compared to externally generated stimuli.

Clinical application: somatoparaphrenia. In certain brain lesions (right parietal), patients deny ownership of their own limb. In the Active Inference framework, somatoparaphrenia is a failure of the corollary discharge to correctly predict proprioceptive feedback from self-generated movements: the limb moves, but the predicted proprioceptive consequence does not match the actual consequence (because the forward model is impaired), leading the patient to infer that the limb must belong to someone else (an alternative hypothesis that minimises prediction error).

0.136 20j. Reinforcement Learning: Mathematical Taxonomy

0.136.1 20j.1 The RL Framework

Reinforcement learning (Sutton and Barto 2018) is the computational study of how agents learn to take actions in an environment to maximise cumulative reward. The formal framework:

A **Markov Decision Process** (MDP) $(\mathcal{S}, \mathcal{A}, P, R, \gamma)$:

- State space $\mathcal{S}$, action space $\mathcal{A}$.
- Transition dynamics $P(s'|s, a)$: probability of transitioning to s' from (s, a).
- Reward function $R(s, a, s') \in \mathbb{R}$: immediate reward.
- Discount factor $\gamma \in [0, 1)$: downweights future rewards.

The agent's goal: find a policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$ that maximises the expected discounted return $G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$.

Value functions. The state-value function $V^\pi(s) = \mathbb{E}_\pi[G_t | S_t = s]$ and the action-value function $Q^\pi(s,a) = \mathbb{E}_\pi[G_t | S_t = s, A_t = a]$ satisfy the **Bellman equations**:

$$V^\pi(s) = \sum_a \pi(a|s) \sum_{s'} P(s'|s,a)[R(s,a,s') + \gamma V^\pi(s')],$$

$$Q^\pi(s,a) = \sum_{s'} P(s'|s,a)[R(s,a,s') + \gamma \sum_{a'} \pi(a'|s') Q^\pi(s',a')].$$

0.136.2 20j.2 Model-Free vs Model-Based RL

Model-free RL learns value functions or policies directly from experience, without building an internal model of the environment. Major algorithms:

Q-learning (Watkins 1989): off-policy, learns $Q^*(s,a) = \max_\pi Q^\pi(s,a)$ by iterating:

$$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma \max_{a'} Q(s',a') - Q(s,a)],$$

where (s,a,r,s') is a transition. Converges to Q^* for tabular MDPs under standard conditions.

SARSA (on-policy): $Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma Q(s',a') - Q(s,a)]$ where $a' \sim \pi(\cdot|s')$. Converges to Q^π (the Q-function of the current policy).

Policy gradient methods (REINFORCE; Williams 1992): directly optimise the policy π_θ by gradient ascent on $J(\theta) = \mathbb{E}_\pi[G_t]$:

$$\nabla_\theta J(\theta) = \mathbb{E}_\pi[G_t \nabla_\theta \log \pi_\theta(A_t|S_t)].$$

The REINFORCE estimator is unbiased but high-variance; **actor-critic** methods reduce variance by replacing G_t with the advantage $A^\pi(s,a) = Q^\pi(s,a) - V^\pi(s)$.

Model-based RL builds an explicit model $\hat{P}(s'|s,a)$ of the environment and uses it for planning (computing the optimal policy from the model). Model-based RL is more sample-efficient (can plan with the model) but requires a good model; model-free RL is more robust to model errors.

Cognitive neuroscience mapping. The distinction between model-free and model-based RL has direct neural correlates (Doll et al., 2012): model-free RL corresponds to habitual behaviour (stimulus-response, controlled by dorsal striatum), while model-based RL corresponds to goal-directed behaviour (planning, controlled by prefrontal cortex and hippocampus). The balance between the two systems is determined by cognitive load, time pressure, and the reliability of the learned model.

0.136.3 20j.3 Deep Reinforcement Learning

Deep Q-Networks (DQN; Mnih et al., 2015) combine Q-learning with deep convolutional networks: $Q(s,a;\theta) \approx Q^*(s,a)$ where θ are the network parameters and s is a raw pixel image. Two stabilisation tricks:

- **Experience replay**: store transitions in a replay buffer and sample minibatches, breaking temporal correlations.

- **Target network**: use a separate, slowly-updated network Q^- for the TD target $r+\gamma \max_{a'} Q^-(s', a')$, preventing oscillations.

DQN achieves superhuman performance on 49 Atari games using only raw pixels and game scores, demonstrating that deep RL can solve complex perception-action problems from scratch.

Proximal Policy Optimization (PPO; Schulman et al., 2017) is currently the most widely used deep RL algorithm: it optimises a clipped surrogate objective that prevents large policy updates:

$$\mathcal{L}^{\text{CLIP}}(\theta) = \mathbb{E}_t\Big[\min\Big(r_t(\theta)\hat{A}_t,\ \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t\Big)\Big],$$

where $r_t(\theta) = \pi_\theta(a_t|s_t)/\pi_{\theta_{\text{old}}}(a_t|s_t)$ is the probability ratio and $\hat{A}_t$ is the advantage estimate. The clip prevents r_t from deviating far from 1, ensuring stable training.

0.137 20i. The Mathematics of Metacognition

0.137.1 20i.1 Confidence as Posterior Probability

Metacognition — thinking about one's own thinking — has been studied extensively in cognitive psychology, but only recently has a rigorous mathematical framework emerged. The key insight (Fleming and Dolan 2012; Maniscalco and Lau 2012): **confidence** in a perceptual or cognitive decision is the posterior probability assigned to the chosen option.

In signal detection theory, after deciding $\hat{s} = +1$ based on a measurement x and criterion c (decide $+1$ if $x > c$), the optimal confidence is:

$$P(\text{correct}|x, \hat{s}=+1) = P(s=+1|x) = \frac{P(x|+1)P(+1)}{P(x|+1)P(+1) + P(x|-1)P(-1)}.$$

This is the Bayesian posterior probability of the chosen option given the sensory evidence. For equal priors: $P(\text{correct}|x) = \sigma(x - c + d'(c - \mu_0/d'))$ where σ is the logistic function.

Meta-d' (Maniscalco and Lau 2012) is a signal-theoretic measure of metacognitive efficiency: how much of the available perceptual information is used by the confidence judgement. Define d' as the perceptual sensitivity and m-d' as the sensitivity of the metacognitive system (inferred from the ROC of the confidence ratings). Metacognitive efficiency is mratio $= m\text{-}d'/d'$; an ideal observer has mratio$= 1$ (the confidence judgment uses exactly as much information as the perceptual decision).

0.137.2 20i.2 The Metacognitive Read-Out Model

The **two-stage model of metacognition** (Fleming et al., 2012; Jang et al., 2011):

- Stage 1: A perceptual decision based on noisy evidence $x \sim \mathcal{N}(d' \cdot s, 1)$.
- Stage 2: A confidence judgment based on the same or additional evidence x', potentially corrupted by additional noise $\eta \sim \mathcal{N}(0, \sigma_m^2)$.

The metacognitive noise σ_m^2 degrades the accuracy of confidence judgments relative to the ideal: mratio $= 1/\sqrt{1 + \sigma_m^2/\sigma_1^2}$ where σ_1^2 is the Stage 1 noise. Metacognitive noise can arise from: limited introspective

access to Stage 1 computations, additional sampling noise in the confidence process, or biased priors over one's own accuracy.

Bayesian metacognition. A fully Bayesian account (Fleming and Daw 2017) treats metacognition as inference about one's own decision process: the agent maintains a prior $P(\text{accuracy})$ (how good am I at this task?) and updates it given each decision and its outcome. This is a hierarchical Bayesian model where the parameters of the Stage 1 noise model are themselves uncertain and inferred:

$$P(d'|\text{decisions and outcomes}) \propto P(\text{outcomes}|d')P(d').$$

High metacognitive ability corresponds to accurate estimation of d'; overconfidence corresponds to overestimating d'; underconfidence to underestimating it.

0.137.3 20i.3 Hierarchical Metacognition and the Self-Model

The concept of a **self-model** (Metzinger 2003; Frith 2007) is the agent's model of its own mental states, capacities, and limitations. In the FEP framework (Section IV), the self-model is part of the generative model: internal states $\boldsymbol{\mu}$ encode not only the state of the world but also the agent's own accuracy, reliability, and biases.

The **active self-model**: the agent actively minimises the free energy of its self-model by gathering evidence about its own performance. This generates **metacognitive monitoring** (checking whether one's answers are likely to be correct) and **metacognitive control** (allocating more resources to uncertain decisions, seeking additional information before committing).

Formally, the self-model contributes precision weights in predictive coding: high self-confidence ($\hat{d}'$ large) corresponds to high precision of the agent's perceptual predictions (downweighting sensory prediction errors); low self-confidence corresponds to low precision (upweighting sensory evidence). The optimal precision setting is determined by $\hat{d}'$: if the agent accurately knows its own perceptual accuracy, it optimally weights sensory evidence vs. prior predictions.

0.138 20h. Optimal Control and the Mathematics of Motor Learning

0.138.1 20h.1 Linear Quadratic Regulators

Linear Quadratic Regulation (LQR) is the classical optimal control solution for linear systems with quadratic cost. For a discrete-time linear system $\mathbf{x}_{t+1} = A\mathbf{x}_t + B\mathbf{u}_t + \mathbf{w}_t$ (where $\mathbf{u}_t$ is the control input and $\mathbf{w}_t \sim \mathcal{N}(0, Q_{\text{noise}})$ is process noise), minimise:

$$J = \mathbb{E}\left[\sum_{t=0}^{T-1} (\mathbf{x}_t^\top Q \mathbf{x}_t + \mathbf{u}_t^\top R \mathbf{u}_t) + \mathbf{x}_T^\top P_T \mathbf{x}_T\right],$$

where $Q \succeq 0$ and $R \succ 0$ are cost matrices penalising large states and large controls respectively.

The optimal control policy is linear and time-varying: $\mathbf{u}_t^* = -K_t \mathbf{x}_t$, where the **feedback gain matrices** K_t are computed by the backward **Riccati recursion**:

$$P_t = Q + A^\top P_{t+1} A - A^\top P_{t+1} B (R + B^\top P_{t+1} B)^{-1} B^\top P_{t+1} A,$$

$$K_t = (R + B^\top P_{t+1} B)^{-1} B^\top P_{t+1} A.$$

The matrices P_t are the solutions to the **discrete-time Riccati equation**. For the infinite-horizon problem ($T = \infty$), the Riccati equation has a unique positive definite steady-state solution P^* (when (A, B) is stabilisable and $(A, Q^{1/2})$ is detectable), and the optimal gain is $K^* = (R + B^\top P^* B)^{-1} B^\top P^* A$.

LQR as the brain's motor controller. Todorov and Jordan (2002) proposed that the nervous system implements a form of LQR to control movements. The "minimum intervention principle": the optimal LQR controller only corrects deviations that affect task performance (deviations in the task-relevant subspace), while leaving deviations in the task-irrelevant subspace uncorrected. This is precisely the UCM hypothesis (Section 25c): the brain controls task-relevant dimensions exactly and allows variability in task-irrelevant dimensions.

0.138.2 20h.2 Stochastic Optimal Control and Hamilton-Jacobi-Bellman

For nonlinear systems or non-quadratic costs, LQR does not apply. The general stochastic optimal control problem is solved by the **Hamilton-Jacobi-Bellman (HJB) equation**. For a continuous-time system $d\mathbf{x} = f(\mathbf{x}, \mathbf{u})dt + \sigma(\mathbf{x})d\mathbf{W}_t$, the optimal value function $V^*(\mathbf{x}, t) = \min_{\mathbf{u}} J(\mathbf{x}, t; \mathbf{u})$ satisfies:

$$-\frac{\partial V^*}{\partial t} = \min_{\mathbf{u}} \left[\ell(\mathbf{x}, \mathbf{u}) + \nabla_{\mathbf{x}} V^* \cdot f(\mathbf{x}, \mathbf{u}) + \frac{1}{2} \operatorname{tr}(\sigma \sigma^\top \nabla_{\mathbf{x}}^2 V^*) \right],$$

where ℓ is the running cost. The optimal control is $\mathbf{u}^*(\mathbf{x}, t) = \arg\min_{\mathbf{u}} [\ell(\mathbf{x}, \mathbf{u}) + \nabla_{\mathbf{x}} V^* \cdot f(\mathbf{x}, \mathbf{u})]$.

The HJB equation is a partial differential equation in $(\mathbf{x}, t)$ space. For LQR, the HJB solution is $V^*(\mathbf{x}, t) = \mathbf{x}^\top P_t \mathbf{x}$, recovering the Riccati equation. For general nonlinear systems, the HJB must be solved numerically (dynamic programming, differential dynamic programming, or iterative LQR).

0.138.3 20h.3 Reward-Modulated Hebbian Plasticity and TD Learning

The connection between optimal control and neural plasticity is provided by **Temporal Difference (TD) learning** (Sutton 1988). TD learning approximates the value function $V^\pi(\mathbf{x}) = \mathbb{E}[\sum_t \gamma^t r_t | \mathbf{x}_0 = \mathbf{x}]$ by iterating:

$$V(\mathbf{x}_t) \leftarrow V(\mathbf{x}_t) + \alpha \underbrace{(r_t + \gamma V(\mathbf{x}_{t+1}) - V(\mathbf{x}_t))}_{\delta_t \text{ (TD error)}}.$$

The **TD error** δ_t is the difference between the actual reward plus the value of the next state and the current value estimate. For the optimal policy, $\delta_t = 0$ at every step (the Bellman equation is satisfied). Non-zero δ_t signals a prediction error that drives the value update.

Schultz, Dayan, and Montague (1997) showed that dopaminergic neurons in the ventral tegmental area (VTA) encode δ_t: they fire more than baseline at unpredicted rewards, fire at baseline for predicted rewards, and are suppressed when a predicted reward is omitted. The dopamine signal is the TD error. Dopamine neurons projecting to the striatum modulate synaptic weights via **reward-modulated Hebbian plasticity**: $\Delta w_{ij} = \alpha \cdot \delta \cdot e_{ij}$, where e_{ij} is an eligibility trace (a running average of pre- and post-synaptic activity). This is the neural implementation of TD learning, connecting the mathematics of optimal control to the biology of synaptic plasticity.

0.139 20f. ACT-R: The Mathematical Core

0.139.1 20f.1 The Declarative Memory Module: Activation Equations

ACT-R (Adaptive Control of Thought – Rational; Anderson 1993, 2007) models cognition as the interaction of a set of modules (declarative memory, procedural memory, perceptual-motor modules) orchestrated by a central production system.

The **activation** of a declarative memory chunk i determines its retrieval probability and time. Activation A_i is a running sum:

$$A_i = B_i + \sum_j W_j S_{ji} + \sum_k P_k M P_{ki} + \varepsilon_i,$$

where B_i is the **base-level activation**, W_j are **source activations** (spreading from current goal elements), S_{ji} is the **associative strength** from source j to chunk i, P_k are **partial-matching** weights, MP_{ki} is the mismatch penalty for slot k of the retrieval request, and $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$ is activation noise.

The **base-level learning equation** models the activation accrued from n prior uses of chunk i at times $t_1, \ldots, t_n$ (measured back from the current time):

$$B_i = \log\left(\sum_{j=1}^{n} t_j^{-d}\right) \approx \log\left(\frac{n}{(1-d)(t-t_c)^d}\right),$$

where $d \in (0,1)$ is the **decay parameter** and t_c is the time of the chunk's creation. The approximation assumes practice trials are uniformly distributed over $[t_c, t]$. The power-law form encodes the **practice effect** (more practice = higher activation) and the **forgetting effect** (higher d = faster decay).

The Bayesian interpretation: B_i estimates the log-prior probability that chunk i is needed. Under the assumption that memory is used when needed, the optimal Bayesian base-level is $\log p_i(\text{needed at time } t) = \log \sum_j t_j^{-d}$ (the log-sum of power-law recency terms). The ACT-R base-level equation is therefore the Bayesian optimal activation for the optimal retrieval of a memory system that learns from experience.

0.139.2 20f.2 Retrieval Time and Threshold

The time to retrieve chunk i (if retrieved) is:

$$RT_i = Ie^{-fA_i},$$

where I is the **imaginal time** (fixed time to initiate a retrieval) and f is the **activation-RT slope**. This is an inverse exponential: higher activation gives shorter retrieval time, with the minimum latency I as $A_i \to \infty$.

A chunk is retrieved only if its activation exceeds the **retrieval threshold** τ: if $A_i < \tau$, the retrieval fails. The probability that chunk i is retrieved given a cue C is:

$$P(\text{retrieve } i) = P(A_i > \tau) = \frac{1}{1 + e^{(\tau - A_i^{\text{expected}})/s\sqrt{2}}},$$

where $s = \sigma\sqrt{2}\pi/3$ (logistic approximation to the normal CDF, where σ is the noise standard deviation). The probability depends on the **signal-to-noise ratio** $(A_i - \tau)/s$: chunks far above threshold are reliably

retrieved; chunks near threshold are retrieved probabilistically.

Fan effect. The associative strength S_{ji} decays with the number of chunks associated with source j (the "fan"): $S_{ji} = S - \log(\text{fan}_j)$, where S is the maximum associative strength. The fan effect predicts that retrieval time increases with the number of concepts associated with the cue — a prediction supported by extensive experimental evidence.

0.139.3 20f.3 The Procedural Module and Production Rules

The procedural module selects production rules (condition-action pairs) to execute at each cognitive cycle. Production k has a **utility** U_k (expected gain from firing) and a **cost** C_k. The probability of selecting production k from the conflict set (all matching productions) is:

$$P(k) = \frac{e^{U_k/t}}{\sum_{j \in \text{conflict}} e^{U_j/t}},$$

where t is the **temperature** parameter (analogous to Boltzmann temperature in softmax). This is a logistic regression-style selection rule: higher utility = higher probability of selection.

The utility is updated by reinforcement learning: after a goal completes at time T, the utilities of all productions fired in the production chain are updated:

$$U_k \leftarrow U_k + \alpha(R - T - U_k),$$

where R is the goal's reward, T is the total time taken, and α is the learning rate. This is the ACT-R analogue of the TD-error update in RL: the utility converges to the expected reward-minus-time for each production in the chain.

0.140 22b. Game Semantics and Ludics

0.140.1 22b.1 Game Semantics for Logic

Game semantics (Hintikka 1968; Lorenzen and Lorenz 1978; Abramsky and Jagadeesan 1992) provides a dialogical interpretation of logical formulas: the truth of a formula φ is defined in terms of the existence of a **winning strategy** for the Proponent ($\exists$) against any strategy of the Opponent ($\forall$) in a two-player game over the formula's components.

For a formula φ in first-order logic, the game $G(\varphi)$ proceeds as:

- $\varphi = \psi_1 \wedge \psi_2$: Opponent chooses $i \in \{1, 2\}$; game continues with ψ_i.
- $\varphi = \psi_1 \vee \psi_2$: Proponent chooses $i \in \{1, 2\}$; game continues with ψ_i.
- $\varphi = \forall x.\, \psi$: Opponent chooses a domain element a; game continues with $\psi[x := a]$.
- $\varphi = \exists x.\, \psi$: Proponent chooses a domain element a; game continues with $\psi[x := a]$.
- $\varphi = \neg\psi$: Roles swap; game continues with ψ.
- φ atomic: Proponent wins iff φ is true in the model.

A formula φ is true in a model $\mathcal{M}$ iff the Proponent has a winning strategy in $G(\mathcal{M}, \varphi)$. This provides a game-theoretic characterisation of classical truth.

For cognitive semantics, game semantics captures **meaning as use**: the meaning of a sentence is the game associated with asserting it, and understanding a sentence means knowing which moves are valid in response to it. This connects linguistic meaning to pragmatics (speech acts, communicative success) and to the game theory of Section IV: an utterance is a move in a cooperative signalling game, and its meaning is the strategy profile that makes communication successful.

0.140.2 22b.2 Pregroup Grammars: Completeness and Computational Complexity

A **pregroup** is a partially ordered monoid $(P, \leq, \cdot, 1)$ in which every element p has a **left adjoint** p^l and a **right adjoint** p^r satisfying:

$$p^l \cdot p \leq 1 \leq p \cdot p^l, \qquad p \cdot p^r \leq 1 \leq p^r \cdot p.$$

These are the pregroup reductions: $p^l \cdot p$ reduces to 1 (the base sentence type), and 1 expands to $p \cdot p^l$. A **pregroup grammar** assigns to each word w a set of types $\tau(w) \subseteq P$; a string $w_1 \cdots w_n$ is grammatical iff there exist types $t_i \in \tau(w_i)$ such that $t_1 \cdots t_n \leq s$ (the sentence type) in the pregroup.

Example 0.140.1 (Transitive sentence)**.** Assign types: noun phrase *np*, sentence s. A transitive verb has type $np^r s np^l$ (takes an NP on the right, an NP on the left, and produces s). For "Alice loves Bob":

$$\underbrace{np}_{\text{Alice}} \cdot \underbrace{np^r s np^l}_{\text{loves}} \cdot \underbrace{np}_{\text{Bob}}.$$

Using right cancellation: $np \cdot np^r = 1$, so $np \cdot np^r s np^l \cdot np = 1 \cdot s \cdot 1 = s$. The string reduces to s: it is grammatical.

Parsing complexity. Membership in a pregroup grammar (deciding whether a string is grammatical) is solvable in $O(n^3)$ time (cubic in string length), matching context-free parsing. Lambek's original syntactic calculus (which pregroups extend) is also NP-hard for certain extensions; the cubic bound for pregroups is a computational advantage.

0.140.3 22b.3 Distributional Compositional Semantics: Extensions

The DisCoCat framework assigns vectors to words and linear maps to grammatical relations. Several extensions have been developed:

Density matrix semantics (Piedeleu et al., 2015) replaces vectors $\mathbf{v} \in V$ with positive semidefinite operators $\rho \in \text{End}(V)$, $\rho \geq 0$, $\text{tr}(\rho) = 1$. This allows ambiguity and context-dependence to be represented as mixed states: an ambiguous word (e.g., "bank") is a density matrix with non-zero off-diagonal terms, while disambiguation by context corresponds to a quantum measurement that collapses the density matrix to a pure state.

Sentence similarity and entailment. In the vector space semantics, sentence similarity is inner product (cosine similarity): $\text{sim}(s_1, s_2) = \mathbf{s}_1^\top \mathbf{s}_2 / (\|\mathbf{s}_1\| \|\mathbf{s}_2\|)$. Textual entailment ($s_1 \vDash s_2$) can be approximated by asymmetric similarity measures such as the **projection length**: $s_1 \vDash s_2 \approx \mathbf{s}_1^\top \mathbf{s}_2 / \|\mathbf{s}_2\|^2$, which is high iff $\mathbf{s}_1$ points mostly in the direction of $\mathbf{s}_2$.

Relational models. In **Concrete Mathematical Structures** (CMS) of Coecke and Kissinger, morphisms in the semantic category correspond to operations on neural data structures. The mathematical language is that of **compact closed categories** with specified concrete implementations in tensor algebra, making the compositional semantics directly amenable to matrix-algebraic computation.

0.141 22g. Formal Pragmatics and Game Theory of Language

0.141.1 22g.1 Gricean Maxims as Bayesian Inference

Grice's cooperative principle (Grice 1975) proposes that speakers are cooperative: they make contributions that are true, informative, relevant, and clear. The maxims of **Quality** (be truthful), **Quantity** (be informative), **Relation** (be relevant), and **Manner** (be clear) together predict a large range of **pragmatic inferences** (implicatures) that go beyond the literal meaning of utterances.

The **Rational Speech Acts** (RSA) model (Frank and Goodman 2012; Goodman and Frank 2016) formalises Gricean pragmatics as Bayesian inference. A **literal listener** L_0 updates beliefs about the world state w based on an utterance u:

$$P_{L_0}(w|u) \propto P(u|w)P(w),$$

where $P(u|w) = 1$ if u is literally true in w and 0 otherwise. A **pragmatic speaker** S_1 chooses utterances based on their informativeness:

$$P_{S_1}(u|w) \propto \exp(\lambda \log P_{L_0}(w|u) - \text{Cost}(u)),$$

where $\lambda > 0$ is an optimality parameter and $\text{Cost}(u)$ penalises longer or less probable utterances. A **pragmatic listener** L_1 inverts the speaker model:

$$P_{L_1}(w|u) \propto P_{S_1}(u|w)P(w).$$

Scalar implicature. For the utterance "some of the apples are red," the literal meaning allows "all apples are red" (since "all" entails "some"). But the pragmatic listener infers that the speaker would have said "all" if all were red (since "all" is more informative); therefore, not all must be red. This **scalar implicature** is a Gricean inference that RSA models capture: $P_{L_1}(\text{some but not all}|\text{"some"}) \gg P_{L_0}(\text{some but not all}|\text{"some"})$.

0.141.2 22g.2 Signalling Games and the Evolution of Meaning

Lewis signalling games (Lewis 1969) model the evolutionary origin of linguistic conventions. Two agents — sender and receiver — play a game: the sender observes a state t from a set of states $\mathcal{T}$, sends a signal $s \in \mathcal{S}$, and the receiver chooses an action $a \in \mathcal{A}$. Both agents receive reward $r(t, a)$ if the action matches the state (a pure coordination game).

A **signalling system** is a pair of strategies (sender strategy $\sigma : \mathcal{T} \to \Delta(\mathcal{S})$ and receiver strategy $\rho : \mathcal{S} \to \Delta(\mathcal{A})$) that achieves perfect coordination: the receiver always takes the correct action. With $|\mathcal{T}| = |\mathcal{S}| = |\mathcal{A}| = n$, there are $n!$ perfect signalling systems (all bijections from states to signals to actions).

Under replicator dynamics, signalling systems are the only evolutionarily stable strategies (ESS): any population that starts near a signalling system converges to it, and is resistant to invasion by mutants. This

explains how linguistic conventions can arise and stabilise without explicit agreement: the coordination game's ESS structure provides the evolutionary pressure for consistent signal-meaning associations.

0.142 22f. Spatial Language and Conceptual Metaphor

0.142.1 22f.1 The Geometry of Spatial Prepositions

Spatial prepositions ("in," "on," "above," "near") encode geometric relationships between objects. A **functional geometric** model (Regier and Carlson 2001; Logan and Sadler 1996) represents the spatial relationship between a **located object** (LO) and a **reference object** (RO) as a function of the angle and distance from the RO's centre to the LO.

For the preposition "above": the acceptability of "the lamp is above the table" depends on the angle θ of the vector from the table's centre to the lamp's centre. The **Angular Deviation** model: $P(\text{"above"}) = f(\cos\theta, d)$ where $\cos\theta$ is the cosine of the angle from vertical and d is the distance. For perfectly vertical alignment ($\theta = 0$), acceptability is high; it decreases as θ increases.

The **Attentional Vector Sum** (AVS) model (Regier and Carlson 2001) averages over all points of the reference object:

$$\mathbf{v}_{\text{LO,RO}} = \sum_{p \in \text{RO}} \text{Attention}(p) \cdot \frac{\mathbf{r}_{\text{LO}} - \mathbf{p}}{|\mathbf{r}_{\text{LO}} - \mathbf{p}|},$$

and computes the acceptability of a preposition from the angle of $\mathbf{v}_{\text{LO,RO}}$ and its magnitude. The attention weights $\text{Attention}(p)$ are higher for prominent parts of the RO (e.g., the top surface of a table is more prominent for "above" than the legs).

0.142.2 22f.2 Conceptual Metaphor Theory and Its Formalisation

Conceptual Metaphor Theory (Lakoff and Johnson 1980; Lakoff and Núñez 2000) proposes that abstract concepts are understood through **metaphorical mappings** from concrete, perceptual domains. For example, the metaphor TIME IS SPACE maps spatial terms to temporal concepts: "looking forward to the future," "falling behind schedule," "a long meeting."

Formally, a conceptual metaphor is a **partial structure-preserving map** ϕ : Source Domain $\to$ Target Domain that:

1. Maps entities: $\phi(\text{spatial location}) = \text{point in time}$.
2. Maps relations: $\phi(\text{left-of}) = \text{before}$, $\phi(\text{right-of}) = \text{after}$.
3. Preserves inferences: if "left-of" is transitive, then "before" should be transitive.

The mapping is partial (not all source domain structure maps to the target) and non-literal (spatial locations are not literally identical to temporal points).

Information-theoretic interpretation. The metaphor ϕ is a **compression**: it maps the rich structure of the target domain (time, which is complex and abstract) to the well-understood source domain (space), compressing the representation. The Minimum Description Length of abstract concepts is reduced by expressing them in terms of concrete spatial concepts: $\text{MDL}(\text{time via space}) < \text{MDL}(\text{time directly})$. This information-theoretic view of metaphor predicts which metaphors are cognitively natural (those

with high compression ratio: many target-domain inferences follow from few source-domain mappings) and which are strained (low compression ratio).

0.143 22e. Compositionality and the Architecture of Meaning

0.143.1 22e.1 The Principle of Compositionality: Mathematical Formulation

Frege's principle of compositionality (Frege 1892, also known as the **Principle of Semantic Composition**): the meaning of a complex expression is a function of the meanings of its parts and their syntactic combination. Formally:

$$[\![\alpha(\beta)]\!] = F([\![\alpha]\!], [\![\beta]\!], \text{syn}(\alpha, \beta)),$$

where F is a **composition function** depending on α's and β's meanings and their syntactic relationship syn.

In the simplest case (Montague semantics), F is functional application: $[\![\alpha(\beta)]\!] = [\![\alpha]\!]([\![\beta]\!])$. The syntactic relationship is absorbed into the type: if α has type $\langle\sigma, \tau\rangle$ and β has type σ, the composition is the unique composition function of that type.

Productive and systematic consequences of compositionality:

- **Productivity**: given a finite lexicon and finite set of composition rules, infinitely many sentences can be formed and interpreted. The number of sentences of length n grows at least exponentially in n for any non-trivial grammar.
- **Systematicity**: if an agent understands sentence $\alpha(\beta)$ (e.g., "John loves Mary"), it also understands $\beta(\alpha)$-like sentences ("Mary loves John"), because understanding requires knowing the meanings of the parts (α and β) and the composition function, and these transfer systematically.

Fodor and Pylyshyn (1988) argued that compositionality and systematicity require a **language of thought** (LOT): a representational system with discrete combinatorial syntax, not just a statistical correlational structure. The LOT hypothesis stands in tension with connectionist approaches, which represent meanings as vectors in a continuous space without discrete combinatorial structure.

0.143.2 22e.2 Tensor Product Representations

Tensor product representations (Smolensky 1990) attempt to reconcile compositionality with connectionism. The idea: represent a complex structure $[r/f]$ (role r filled by filler f, e.g., the subject role filled by "John") as the tensor product $\mathbf{f} \otimes \mathbf{r}$ of the filler vector $\mathbf{f}$ and role vector $\mathbf{r}$. A structured object is the sum of its tensor-product constituents:

$$\text{rep}(\text{John loves Mary}) = \mathbf{f}_{\text{John}} \otimes \mathbf{r}_{\text{subj}} + \mathbf{f}_{\text{loves}} \otimes \mathbf{r}_{\text{verb}} + \mathbf{f}_{\text{Mary}} \otimes \mathbf{r}_{\text{obj}}.$$

The binding between filler and role is encoded by the tensor product; different bindings are represented in orthogonal subspaces (if $\mathbf{r}_{\text{subj}} \perp \mathbf{r}_{\text{obj}}$).

Unbinding. Given the tensor product representation and a role vector $\mathbf{r}$, the filler is recovered by inner product: $\mathbf{f} \approx \text{rep} \cdot \mathbf{r} = (\mathbf{f} \otimes \mathbf{r}) \cdot \mathbf{r} = \mathbf{f}\|\mathbf{r}\|^2 \approx \mathbf{f}$ (for unit $\mathbf{r}$). Unbinding is the tensor-product analogue of variable access in a symbolic system.

The tensor product representation has several properties:

- **Compositionality**: the representation of the complex is a function (the sum of tensor products) of the representations of the parts and their roles.
- **Context-sensitivity**: the same filler in different roles has different representations ($\mathbf{f}_{\text{John}} \otimes \mathbf{r}_{\text{subj}} \neq \mathbf{f}_{\text{John}} \otimes \mathbf{r}_{\text{obj}}$).
- **Approximate unbinding**: due to vector dimensions and orthogonality approximations, unbinding introduces noise; perfect unbinding requires exact orthogonality of role vectors, which limits capacity to $O(d)$ bindings.

Holographic Reduced Representations (HRR; Plate 1995) improve on tensor products by using circular convolution instead of tensor product for binding. The convolution $\mathbf{f} \circledast \mathbf{r}$ (point-wise multiplication in the frequency domain, inverse FFT) has the same dimension as $\mathbf{f}$ and $\mathbf{r}$, avoiding the quadratic growth of tensor products. Unbinding uses circular correlation: $\mathbf{f} \approx \text{rep} \circledast^{-1} \mathbf{r}$. HRR can represent structures of arbitrary depth with fixed-dimensional vectors, at the cost of approximate (rather than exact) unbinding.

0.143.3 22e.3 Cognitive Architectures for Language: From RNNS to Transformers

Recurrent neural networks (RNNs) and Transformers implement compositional computations in different ways, with different implications for cognitive plausibility.

RNNs as automata. The Elman network (1990) processes a sentence left-to-right, updating a hidden state $\mathbf{h}_t = f(W_h \mathbf{h}_{t-1} + W_x \mathbf{x}_t)$ at each word. Elman found that the hidden states of a trained network cluster according to grammatical category and semantic similarity: the network "discovers" linguistic structure from distributional statistics. However, simple RNNs (tanh, sigmoid) fail on long-range dependencies: the gradient signal decays exponentially over time (the **vanishing gradient problem**), making it difficult to learn dependencies spanning more than ~ 10 tokens.

LSTMs and gating. Long Short-Term Memory networks (Hochreiter and Schmidhuber 1997) address the vanishing gradient problem by introducing a **cell state** $\mathbf{c}_t$ updated by gated additions and multiplications:

$$\mathbf{f}_t = \sigma(W_f[\mathbf{h}_{t-1}, \mathbf{x}_t] + b_f), \quad \mathbf{i}_t = \sigma(W_i[\mathbf{h}_{t-1}, \mathbf{x}_t] + b_i), \quad \tilde{\mathbf{c}}_t = \tanh(W_c[\mathbf{h}_{t-1}, \mathbf{x}_t] + b_c),$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{c}}_t, \quad \mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t).$$

The forget gate $\mathbf{f}_t$ controls how much of the previous cell state to retain; the input gate $\mathbf{i}_t$ controls how much of the new candidate $\tilde{\mathbf{c}}_t$ to include; the output gate $\mathbf{o}_t$ controls what to expose to the hidden state. The constant error carousel (the unimpeded gradient flow through the cell state addition) prevents gradient vanishing over long dependencies.

Transformers and global attention. The Transformer replaces sequential processing with global attention: all positions attend to all positions simultaneously. The attention mechanism computes, for each position i, a weighted sum of all position representations, with weights given by the softmax of scaled inner products. This enables the model to directly compute dependencies between any two positions in $O(1)$ steps (rather than $O(n)$ for RNNs), at the cost of $O(n^2)$ memory for the attention matrix.

0.144 22d. Montague Semantics: Complete Treatment

0.144.1 22d.1 Intensional Logic and Possible Worlds

Montague semantics (Montague 1970, 1973) revolutionised formal linguistics by treating natural language with the same mathematical rigour as formal logic. Its core insight: the meaning of a sentence is not simply a truth value but a **function from possible worlds to truth values** (an **intension**), or more generally, a function from indices (world-time pairs) to extensions.

The semantic types form a hierarchy:

- **Basic types:** e (entities), t (truth values).
- **Function types:** If σ and τ are types, then $\langle \sigma, \tau \rangle$ is the type of functions from σ-denotations to τ-denotations.
- **Intensional types:** If σ is a type, then $s\sigma$ is the type of functions from possible worlds W to σ-denotations (intensions of type σ).

The denotation of an expression of type τ at world w and time t is written $[\![\alpha]\!]^{M,w,t}$ (evaluated in model M at index (w, t)). Key denotations:

- A **proper name** like "John" has type e: $[\![\text{John}]\!] = \mathbf{j}$ (a specific entity).
- A **common noun** like "student" has type $\langle e, t \rangle$: $[\![\text{student}]\!] = \lambda x.\text{student}(x)$ (a characteristic function).
- A **transitive verb** like "loves" has type $\langle e, \langle e, t \rangle \rangle$: $[\![\text{loves}]\!] = \lambda y \lambda x.\text{loves}(x, y)$.
- A **quantifier** like "every" has type $\langle \langle e, t \rangle, \langle \langle e, t \rangle, t \rangle \rangle$: $[\![\text{every}]\!] = \lambda P \lambda Q.\forall x(P(x) \Rightarrow Q(x))$.

The meaning of a sentence is computed by **Functional Application** (FA): the meaning of $[\alpha \ \ \beta]$ is $[\![\alpha]\!]([\![\beta]\!])$, provided $[\![\alpha]\!]$ has a function type whose domain matches the type of $[\![\beta]\!]$. The type calculus is the simply-typed lambda calculus, and well-formedness is guaranteed by type-checking.

0.144.2 22d.2 Quantifier Raising and LF

Quantifiers pose a fundamental challenge for direct compositional interpretation. In "Every student passed," the quantifier "every student" of type $\langle \langle e, t \rangle, t \rangle$ needs to combine with the VP "passed" of type $\langle e, t \rangle$. This is fine. But in "A teacher graded every student," the subject "a teacher" and object "every student" interact in complex ways (scope ambiguity: every student takes scope over $\exists$ teacher, or vice versa).

Quantifier Raising (QR) is the syntactic operation that moves quantifiers to their scope position, creating a **Logical Form** (LF) representation that is interpretable compositionally. The two LFs for the ambiguous sentence are:

- Wide scope for "every student": [every student$_1$ [a teacher$_2$ [t_2 graded t_1]]], giving $\forall y(\text{student}(y) \Rightarrow \exists x(\text{teacher}(x) \wedge \text{graded}(x, y)))$.
- Wide scope for "a teacher": [a teacher$_2$ [every student$_1$ [t_2 graded t_1]]], giving $\exists x(\text{teacher}(x) \wedge \forall y(\text{student}(y) \Rightarrow \text{graded}(x, y)))$.

The type-logical framework handles this via **type raising** and **composition**: the subject NP of type e is raised to type $\langle\langle e, t\rangle, t\rangle$ by the type-raising rule $e \mapsto \langle\langle e, t\rangle, t\rangle$: $\mathbf{j} \mapsto \lambda P.P(\mathbf{j})$. This allows the NP to compose with the VP in a uniform way regardless of whether the subject is a name or a quantifier.

0.144.3 22d.3 Dynamic Semantics and Discourse

Dynamic Predicate Logic (DPL; Groenendijk and Stokhof 1991) and **Dynamic Montague Grammar** extend Montague semantics to handle **discourse anaphora**: the phenomenon where a pronoun ("he," "it") can refer to an antecedent introduced in a previous sentence.

In static logic, "A man walked in. He sat down" is problematic: the pronoun "he" refers to the existentially quantified man of the first sentence, but the existential's scope does not extend to the second sentence. DPL resolves this by making the **meaning of a formula a relation between input and output assignment functions**:

$$[\![\varphi]\!] = \{(g, h) : h \text{ is an extension of } g \text{ satisfying } \varphi\}.$$

The existential $\exists x\, \varphi$ updates the assignment: any h that satisfies φ has a value for x, which is then available for subsequent pronouns. Composition of meanings is relational composition: $[\![\varphi; \psi]\!] = [\![\varphi]\!] \circ [\![\psi]\!]$.

The cognitive significance: dynamic semantics models the mental **discourse representation structure** (DRS) of Kamp (1981). As a text is processed, a DRS is built up incrementally, with new entities introduced by indefinite NPs and accessed by subsequent pronouns. The mathematical structure of DRS is a partial function (assignment) updated by the dynamic semantic rules, and the entire discourse meaning is the final DRS after processing all sentences.

0.145 22c. Type-Theoretic Semantics and Proof-Theoretic Meaning

0.145.1 22c.1 Propositions as Types: The Curry-Howard Correspondence

The **Curry-Howard correspondence** (Curry 1934; Howard 1980) identifies propositions with types and proofs with programs:

Logic	Type Theory
Proposition A	Type A
Proof of A	Term of type A
$A \Rightarrow B$	Function type $A \to B$
$A \wedge B$	Product type $A \times B$
$A \vee B$	Sum type $A + B$
$\forall x : A.\, B(x)$	Dependent product type $\Pi_{x:A} B(x)$
$\exists x : A.\, B(x)$	Dependent sum type $\Sigma_{x:A} B(x)$

Under this correspondence, a proof of $A \Rightarrow B$ is a function $f : A \to B$ that converts any proof of A into a proof of B. The logical rule **modus ponens** corresponds to **function application**: from $f : A \to B$ and $a : A$, derive $f(a) : B$.

Dependent type theory (Martin-Löf 1975) extends this to dependent types, where the type $B(x)$ may depend on the value $x : A$. For example, the type "vectors of length n" is $\mathrm{Vec}(A, n)$ for $n : \mathbb{N}$; the type

depends on the value n. Dependent types allow precise specification: the type of "matrix multiplication" is $\Pi_{m,n,p:\mathbb{N}}\text{Mat}(m,n) \to \text{Mat}(n,p) \to \text{Mat}(m,p)$, which encodes the compatibility constraint on matrix dimensions.

For linguistics, the Curry-Howard correspondence suggests that sentences are types and their interpretations are proofs. A well-formed sentence is one whose type reduces (through syntactic derivation) to the sentence type s; the meaning of the sentence is the proof term that witnesses this reduction. This is the **proof-theoretic semantics** tradition (Dummett, Prawitz), which grounds meaning in the verification or use conditions of sentences rather than in model-theoretic truth conditions.

0.145.2 22c.2 Homotopy Type Theory and Cognitive Representations

Homotopy Type Theory (HoTT; Voevodsky 2010; Univalent Foundations 2013) extends Martin-Löf type theory with the **univalence axiom**: two types A and B are equal iff they are equivalent (i.e., there is a homotopy equivalence $A \simeq B$). Under this axiom, the equality type $a =_A b$ for $a, b : A$ is interpreted as a **path** from a to b in the space A, making the type theory into a formal language for algebraic topology.

For cognitive science, HoTT provides a principled account of **conceptual equivalence**: two representations are conceptually equivalent iff they are connected by a path in the type-theoretic space of representations. This formalises the cognitive linguist's intuition that concepts form a **conceptual space** (Gärdenfors 2000) with a geometric structure: similar concepts are close in the space, and conceptual change follows paths through the space.

The practical consequence: if the brain implements a type-theoretic representation of knowledge, then **conceptual generalisation** corresponds to lifting: given a computation $f : A \to B$ and a path $p : a =_A a'$, the **action** of f on p is a path $\text{ap}_f(p) : f(a) =_B f(a')$. Any function that maps concepts to concepts automatically maps conceptual changes to conceptual changes, implementing a form of systematic generalisation that is provably consistent with the type structure.

0.145.3 22c.3 Vector Space Models: Empirical Validation

The theoretical DisCoCat framework connects to an extensive empirical literature on distributional semantics. **Word vectors** (word embeddings) represent words as high-dimensional vectors based on their co-occurrence statistics; the geometry of the vector space captures semantic relationships.

The **Word2Vec** model (Mikolov et al., 2013) learns word vectors by training a neural network to predict context words from a target word (skip-gram) or vice versa (CBOW). The learned vectors have the empirical property that semantic relations correspond to vector differences: $\mathbf{v}(\text{king}) - \mathbf{v}(\text{man}) + \mathbf{v}(\text{woman}) \approx \mathbf{v}(\text{queen})$.

GloVe (Pennington et al., 2014) derives this from the statistical structure of co-occurrence matrices. Define X_{ij} as the number of times word j occurs in the context of word i. The GloVe objective is:

$$J = \sum_{i,j=1}^{V} f(X_{ij})(\mathbf{w}_i^\top \tilde{\mathbf{w}}_j + b_i + \tilde{b}_j - \log X_{ij})^2,$$

where $f(x) = \min(1, (x/x_{\max})^\alpha)$ is a weighting function that downweights frequent pairs. Minimising J finds word vectors $\mathbf{w}_i$ and context vectors $\tilde{\mathbf{w}}_j$ such that $\mathbf{w}_i^\top \tilde{\mathbf{w}}_j \approx \log P(j|i) - \log P(j)$ (the pointwise

mutual information, PMI). Semantic analogy tasks reduce to vector arithmetic: $\mathbf{v}(a : b) \approx \mathbf{v}(c : d)$ iff $\mathbf{v}(b) - \mathbf{v}(a) \approx \mathbf{v}(d) - \mathbf{v}(c)$.

The connection to DisCoCat: word vectors are the concrete implementations of the abstract vector space semantics. The DisCoCat functor assigns to each word a vector or tensor in a finite-dimensional semantic space; the GloVe/Word2Vec vectors provide the data-driven values of these tensors. The functor structure then assembles word vectors into sentence vectors via tensor contraction — precisely the compositional computation described in Section 23.

Limitations. Distributional vectors capture *syntagmatic* and *paradigmatic* relations well (words that occur in similar contexts, words that substitute for each other) but struggle with *logical* structure (negation, quantifiers, conditionals). The vector representation of "not happy" is typically closer to "happy" than to synonyms of "unhappy" — the geometry does not naturally encode logical negation. This gap between distributional and logical semantics is an active research frontier, partly addressed by neuro-symbolic approaches that combine continuous vector representations with discrete logical operations.

Section *XII*

Section XII — Embodied Cognition and Mathematical Limits

0.146 24. The Body in the Mind: Beyond the Representationalist Paradigm

The preceding Section s have developed mathematical frameworks that, with the exception of Active Inference, treat cognition primarily as information processing: inference over distributions, computation over representations, reasoning in formal systems. This is a powerful and productive perspective. It is also incomplete.

The **embodied cognition** movement, drawing on phenomenology (Merleau-Ponty), enactivism (Varela, Thompson, Rosch), and ecological psychology (Gibson), argues that cognition is not just computation in the brain but a dynamical process involving the brain, the body, and the environment in real time. Thinking is not merely looking things up in a mental database; it is an ongoing sensorimotor loop in which action and perception are inextricably coupled.

This Section develops the mathematics of embodied cognition: dynamical systems theory as the formal language for agent-environment coupling, the mathematics of sensorimotor contingencies, and the limits that embodiment imposes on any purely symbolic or probabilistic account.

0.147 25b. Impossibility Theorems and the Limits of Formalisation

0.147.1 25b.1 Gödel's Incompleteness and the Limits of Mathematical Cognition

Gödel's First Incompleteness Theorem (1931) states that any consistent formal system F that is sufficiently strong (capable of representing basic arithmetic) contains a sentence G_F that is true (in the standard model of arithmetic) but not provable in F. The sentence G_F is the formalisation of "I am not provable in F."

The proof uses **arithmetisation**: Gödel encodes formulas and proofs as natural numbers (via Gödel numbering), then constructs a formula $\text{Prov}_F(n)$ that is true iff n is the Gödel number of a formula provable in F. The Gödel sentence G_F is constructed by the diagonal lemma: there exists a sentence G_F such that $F \vdash G_F \leftrightarrow \neg\text{Prov}_F(\ulcorner G_F \urcorner)$.

If F is consistent and G_F is provable in F, then $\neg\text{Prov}_F(\ulcorner G_F \urcorner)$ is also provable (by the proof just given), contradicting consistency. If $\neg G_F$ is provable, then $\text{Prov}_F(\ulcorner G_F \urcorner)$ is provable, meaning G_F would be provable too, again a contradiction. Therefore neither G_F nor $\neg G_F$ is provable: G_F is undecidable.

For cognitive science, the philosophical implications of incompleteness are contested. **Penrose** (1989, 1994) argued that human mathematical understanding transcends formal systems, because humans can "see" that G_F is true even though F cannot prove it. This argument (the Penrose-Lucas argument) has been criticised on multiple grounds: (i) knowing G_F is true requires knowing F is consistent, which is not given; (ii) humans do not directly "see" the truth of Gödel sentences; (iii) the argument conflates the mathematician's informal understanding with the formal system being described. The mathematical community's consensus is that incompleteness constrains formal systems, not human cognition per se.

Gödel's Second Incompleteness Theorem is directly relevant: no consistent system F can prove its own consistency, $\text{Con}(F)$. Cognitively, this implies that any cognitive agent operating by formal rules cannot, from within its own rules, guarantee the coherence of those rules. External validation always requires a "stronger" meta-system—a regress that Wittgenstein, from a different direction, also identified as foundational to the practice of rule-following.

0.147.2 25b.2 The Frame Problem: Formal Analysis

The **Frame Problem** (McCarthy and Hayes 1969) asks: how can a reasoning agent efficiently represent and update what does *not* change when an action is performed? In a world with many facts, most facts are unaffected by any given action, but a formal system must be told this explicitly for each fact and action pair—exponentially many "non-effects" clauses.

Formally, in **situation calculus**, facts are expressed as $\text{Holds}(f, s)$ (fluent f holds in situation s). The effect of action a in situation s is situation $\text{do}(a, s)$. Without a frame axiom, we would need to assert for every fluent f not affected by action a:

$$\forall f, s: \ [\text{AffectedBy}(f, a) \vee (\text{Holds}(f, s) \Rightarrow \text{Holds}(f, \text{do}(a, s)))].$$

With many fluents and actions, the number of such axioms is $|\text{fluents}| \times |\text{actions}|$.

The **Successor State Axiom** approach (Reiter 1991) resolves this by providing a single axiom for each fluent f that specifies exactly the conditions under which f changes:

$$\text{Holds}(f, \text{do}(a, s)) \Leftrightarrow [\gamma_f^+(a, s) \vee (\text{Holds}(f, s) \wedge \neg\gamma_f^-(a, s))],$$

where γ_f^+ encodes when a makes f true and γ_f^- encodes when a makes f false. This requires one axiom per fluent (rather than per fluent-action pair), reducing the axiom count from $O(FA)$ to $O(F)$.

For probabilistic planning, the frame problem becomes the **relevance problem**: given a state with n fluents and an action with expected effects on k of them, the agent must update beliefs about all n fluents, but only k require recomputation. In MDPs and POMDPs, this is handled by **factored representations** (dynamic Bayesian networks, factored MDPs): the transition model factorises over independent groups of state variables, and belief updating requires only local propagation.

0.147.3 25b.3 Coupled Dynamical Systems: Stability and Bifurcations

The agent-environment coupling $(\dot{\mathbf{a}}, \dot{\mathbf{e}}) = (f_A(\mathbf{a}, \mathbf{e}), f_E(\mathbf{e}, \mathbf{a}))$ of Section 25 can be analysed using bifurcation theory. The system's qualitative behaviour—the number and stability of fixed points, limit cycles, and chaotic attractors—depends on a finite number of parameters, and it can change discontinuously at **bifurcation points**.

The **saddle-node bifurcation** is the generic mechanism for the sudden appearance or disappearance of equilibria. Near the bifurcation point μ_c, the dynamics reduce to the normal form:

$$\dot{x} = \mu - x^2.$$

For $\mu < 0$: no fixed points; for $\mu > 0$: two fixed points $x^* = \pm\sqrt{\mu}$ (one stable, one unstable). The saddle-node captures the phenomenology of perceptual multistability (bistable perception, where the percept flips between two stable states) and the onset of motor programs (sudden transition from one movement pattern to another).

The **Hopf bifurcation** produces oscillations. Near the bifurcation: $\dot{x} = (\mu + i\omega)x - |x|^2 x$ in complex form. For $\mu < 0$: stable fixed point; for $\mu > 0$: stable limit cycle of amplitude $\sqrt{\mu}$. Neural oscillations in the gamma band (30–80 Hz), proposed as binding mechanisms in the theory of feature binding by

synchrony, can be modelled as Hopf bifurcations in the dynamics of cortical columns.

The mathematical implication for embodied cognition: cognitive behaviours are not sequences of discrete symbolic operations but **trajectories on an attractor landscape**, continuously shaped by the coupling with the environment. Reasoning, decision-making, and language processing are attractors of the coupled brain-body-environment system, not abstract computations executed on a substrate.

0.148 23c. Experimental Tests of Compositional Semantics

0.148.1 23c.1 Semantic Similarity: Benchmarks and Results

Compositional semantic models are evaluated on **semantic similarity** benchmarks: pairs of sentences rated by human judges for semantic relatedness. Standard benchmarks include:

- **STS-B** (Semantic Textual Similarity Benchmark; Cer et al., 2017): 8,628 sentence pairs with human similarity ratings $[0, 5]$.
- **SICK** (Sentences Involving Compositional Knowledge; Marelli et al., 2014): 10,000 sentence pairs with relatedness and entailment annotations.
- **SimLex-999** (Hill et al., 2015): 999 word pairs with human similarity ratings, designed to distinguish similarity from association.

For compositional models, the predicted similarity between sentences $\mathbf{s}_1$ and $\mathbf{s}_2$ is $\text{sim} = \mathbf{s}_1^\top \mathbf{s}_2 / (\|\mathbf{s}_1\| \|\mathbf{s}_2\|)$ (cosine similarity). Performance is measured by the **Spearman correlation** ρ_s between the model's similarity scores and human ratings. State-of-the-art results (BERT, Sentence-BERT): $\rho_s \approx 0.90$ on STS-B; DisCoCat-style models with full tensor product representations: $\rho_s \approx 0.78$–0.84, achieving competitive performance while providing interpretable compositional structure.

The semantic similarity task reveals a fundamental challenge for purely compositional models: **pragmatic enrichment**. Human similarity ratings for "The bank by the river" and "The financial institution by the water" diverge dramatically from purely compositional predictions because humans enrich both with contextual inference (the disambiguation of "bank" and the metonymy of "water" for "river"). Compositional models handle this poorly; distributional models that encode collocational information handle it better.

0.148.2 23c.2 Entailment and Recognising Textual Entailment

Natural Language Inference (NLI; Dagan et al., 2006) is the task of determining whether a **hypothesis** H is entailed by, contradicts, or is neutral w.r.t. a **premise** P. Benchmark datasets: SNLI (570k examples), MultiNLI (433k examples), and HANS (probing compositional reasoning).

Compositional models for NLI encode both P and H into vector representations and classify their relationship. The key compositional challenge: negation. The DisCoCat representation of "John did not leave" and "John left" should be opposite in some semantic direction; the tensor product model handles this by learning a negation tensor $\text{neg} \in V \otimes V^*$ that maps the truth value dimension of the sentence to its negation.

Formally, for a sentence $\mathbf{s} = f(\text{words})$ and its negation $\neg\mathbf{s} = \text{neg}(\mathbf{s})$, the compositional model requires neg to implement semantic negation: $\mathbf{s}^\top \mathbf{s}_+ > 0$ and $(\text{neg}(\mathbf{s}))^\top \mathbf{s}_+ < 0$ for any positive context $\mathbf{s}_+$. Learn-

ing the negation tensor from NLI data is possible when negative examples are present; the mathematical structure constrains the solution to be an involution: $\text{neg}(\text{neg}(\mathbf{s})) = \mathbf{s}$.

0.149 23b. Category Theory of Meaning: Monoidal Structures

0.149.1 23b.1 Monoidal Categories and Tensor Products

A **monoidal category** $(\mathcal{C}, \otimes, I, \alpha, \lambda, \rho)$ is a category with:

- A bifunctor $\otimes : \mathcal{C} \times \mathcal{C} \to \mathcal{C}$ (the **tensor product**).
- A unit object I.
- Natural isomorphisms: $\alpha_{A,B,C} : (A \otimes B) \otimes C \cong A \otimes (B \otimes C)$ (**associator**), $\lambda_A : I \otimes A \cong A$ (**left unitor**), $\rho_A : A \otimes I \cong A$ (**right unitor**).
- Coherence conditions (Mac Lane's pentagon and triangle identities).

A monoidal category is **symmetric** if additionally $\sigma_{A,B} : A \otimes B \cong B \otimes A$ (the **symmetry**). The category **FVect** of finite-dimensional vector spaces with the tensor product $\otimes$ (linear-algebraic tensor product) and unit $\mathbb{R}$ is the canonical symmetric monoidal category for compositional semantics.

A **compact closed category** (Abramsky and Coecke 2004) is a symmetric monoidal category where every object A has a **dual** A^* with morphisms:

$$\eta_A : I \to A^* \otimes A \quad \text{(unit)}, \qquad \epsilon_A : A \otimes A^* \to I \quad \text{(counit)},$$

satisfying the **snake equations**: $(\epsilon_A \otimes \text{id}_A) \circ (\text{id}_A \otimes \eta_A) = \text{id}_A$ and symmetrically. In **FVect**, the dual $A^* = \text{Hom}(A, \mathbb{R})$ is the dual space, and the unit and counit are the canonical maps $\eta : \mathbb{R} \to A^* \otimes A$ and $\epsilon : A \otimes A^* \to \mathbb{R}$ (inner product).

String diagrams are the graphical calculus of compact closed categories: morphisms are drawn as boxes, objects as wires, tensor product is juxtaposition, composition is vertical stacking, and the snake equations are depictions of cups and caps that can be slid along wires. String diagrams provide a powerful visual intuition: disentangling a diagram by isotopy (continuous deformation without cutting wires) proves equalities between morphisms.

0.149.2 23b.2 The DisCoCat Functor as a Semantics

The **DisCoCat** (Distributional Compositional Categorical; Coecke, Sadrzadeh, Clark 2010) framework defines meaning as a **functor** F : **Grammar** $\to$ **Meaning** from a compact closed category of grammatical types to a compact closed category of meaning vectors.

The **pregroup grammar** (Section 22.1) forms a compact closed category **Preg**: objects are pregroup types (n, s, n^l, n^r, ...), morphisms are derivations (sequences of left and right reductions), and the tensor product is the concatenation of types.

The functor F : **Preg** $\to$ **FVect** assigns:

- $F(n) = N$ (noun space, $\mathbb{R}^{d_N}$).

- $F(s) = S$ (sentence space, $\mathbb{R}^{d_S}$, often $d_S = 1$: a real number representing truth value, or $d_S = d_N$ for full sentence meaning).
- $F(n^l) = F(n^r) = N^* \cong N$ (in **FVect**, $V^* \cong V$ for finite-dimensional spaces with inner product).
- $F(\text{composition}) = \text{tensor contraction}$.

For a transitive sentence "Alice loves Bob" with derivation $n \cdot (n^r s n^l) \cdot n \to s$:

$$F(\text{sentence}) = F(\text{loves})(F(\text{Alice}), F(\text{Bob})),$$

where $F(\text{loves}) \in N \otimes S \otimes N^*$ is the tensor for the verb (a trilinear form), and the evaluation is the partial application:

$$\langle F(\text{Alice}), F(\text{loves})(\cdot, F(\text{Bob}))\rangle = \sum_{i,j,k} [\text{Alice}]_i [\text{loves}]_{ijk} [\text{Bob}]_k e_j \in S.$$

This is a bilinear form on noun vectors, producing a sentence vector. The functor F maps the pregroup derivation to a tensor network computation: the grammatical reduction is implemented as tensor contraction, and the meaning of the sentence is the scalar (or vector) result of this contraction.

0.149.3 23b.3 Lambek Calculus and Type-Logical Grammar

The **Lambek calculus L** (Lambek 1958) is a sequent calculus for the type-logical analysis of natural language. It has three inference rules beyond the identity:

Product introduction and elimination:

$$\frac{\Gamma \vdash A \quad \Delta \vdash B}{\Gamma, \Delta \vdash A \cdot B} \cdot I, \qquad \frac{\Gamma, A, B, \Delta \vdash C}{\Gamma, A \cdot B, \Delta \vdash C} \cdot E.$$

Right division introduction and elimination:

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash B/A} /I, \qquad \frac{\Gamma \vdash B/A \quad \Delta \vdash A}{\Gamma, \Delta \vdash B} /E.$$

Left division introduction and elimination:

$$\frac{A, \Gamma \vdash B}{\Gamma \vdash A \backslash B} \backslash I, \qquad \frac{\Delta \vdash A \quad \Gamma \vdash A \backslash B}{\Delta, \Gamma \vdash B} \backslash E.$$

The Lambek calculus is a **non-commutative multiplicative linear logic**: it respects the order of types in a sequent (unlike classical or intuitionistic logic, where $A, B \vdash C$ and $B, A \vdash C$ are equivalent). Word order is encoded in the type derivation.

Theorem 0.149.1 (Cut Elimination; Lambek 1958)**.** In the Lambek calculus, every proof can be normalised to a cut-free proof: any inference using the cut rule $\frac{\Gamma \vdash A \quad A, \Delta \vdash B}{\Gamma, \Delta \vdash B}$ can be eliminated without changing the provability of the sequent.

Cut elimination is the mathematical expression of the **compositionality principle**: the meaning of a complex expression can be computed from the meanings of its parts without appeal to the global context.

The cut rule would allow the meaning of one constituent to "bleed into" another; its elimination proves that this is never necessary.

Completeness. The Lambek calculus is complete for **residuated lattices**: for any sequent not derivable in **L**, there exists a residuated lattice model in which the sequent fails. The cut-free completeness proof (Lambek 1958) proceeds by constructing the Lindenbaum algebra of the calculus and showing it is a residuated lattice.

0.150 25c. The Geometry of Action and Sensorimotor Space

0.150.1 25c.1 Motor Synergies and Dimensionality Reduction

The human motor system controls hundreds of degrees of freedom (joints, muscles) to produce highly reliable movements. The **motor synergy** hypothesis (Bernstein 1967; Latash 2008) proposes that the CNS reduces this complexity by coupling degrees of freedom into low-dimensional **synergies**: coordinated activations of groups of muscles that are controlled as single units.

Mathematically, a synergy is a vector $\mathbf{s} \in \mathbb{R}^M$ (where M is the number of muscles) such that a motor command is expressed as a linear combination $\mathbf{u} = \sum_k c_k \mathbf{s}_k + \mathbf{n}$ where c_k are synergy weights and $\mathbf{n}$ is within-synergy variability (noise in the null space of the task). The synergy matrix $S = [\mathbf{s}_1, \ldots, \mathbf{s}_K]$ with $K \ll M$ defines the low-dimensional **synergy subspace**.

Uncontrolled Manifold (UCM) hypothesis (Scholz and Schöner 1999): the CNS controls only the task-relevant components of motion, allowing unconstrained variability in the task-irrelevant null space. For a task function $\mathbf{f} : \mathbb{R}^M \to \mathbb{R}^d$ (e.g., endpoint position in $\mathbb{R}^3$ from M joint angles), the Jacobian $J = \partial \mathbf{f}/\partial \mathbf{q}$ has a null space $\mathcal{N}(J)$ (all joint-angle variations that do not affect the endpoint) and a task-relevant subspace $\mathcal{R}(J^\top)$.

The UCM prediction: variability in $\mathcal{N}(J)$ (along the uncontrolled manifold) should be larger than variability in $\mathcal{R}(J^\top)$ (orthogonal to the manifold, directly affecting the task). Define:

$$\text{UCM} = \frac{\text{Var}_{\parallel}}{\dim \mathcal{N}(J)/(M-d)}, \qquad \text{ORT} = \frac{\text{Var}_{\perp}}{\dim \mathcal{R}(J^\top)/d},$$

where $\text{Var}_{\parallel}$ and $\text{Var}_{\perp}$ are the variances within and orthogonal to $\mathcal{N}(J)$ respectively. UCM/ORT > 1 indicates that more variability occurs along the uncontrolled manifold than orthogonal to it, consistent with the hypothesis.

0.150.2 25c.2 Riemannian Geometry of Musculoskeletal Systems

The musculoskeletal system defines a Riemannian manifold structure on the configuration space $\mathcal{Q}$ of joint angles. The **inertia tensor** $M(\mathbf{q}) \in \mathbb{R}^{n \times n}$ (where n is the number of degrees of freedom) defines a Riemannian metric: the kinetic energy of a movement is $T = \frac{1}{2}\dot{\mathbf{q}}^\top M(\mathbf{q})\dot{\mathbf{q}}$, and geodesics of the metric correspond to minimum-jerk trajectories in inertia-weighted configuration space.

The **Newton-Euler equations** of motion for a rigid-body system are:

$$M(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + G(\mathbf{q}) = \boldsymbol{\tau},$$

where $C(\mathbf{q}, \dot{\mathbf{q}})$ is the Coriolis-centripetal term, $G(\mathbf{q})$ is the gravitational term, and $\boldsymbol{\tau}$ is the vector of joint torques. The term $C = \frac{1}{2}\dot{M} - \frac{1}{2}(\partial M/\partial q_k \dot{q}_k)_k$ (the Christoffel symbols of the Riemannian metric).

The **Geodesic Hypothesis** of motor control (Flash and Sejnowski 2001; Hogan and Sternad 2007) proposes that the motor system plans movements as geodesics in an appropriate Riemannian metric, where the metric encodes the cost of movement (energy, jerk, variability). Under the task-space metric $g_{ij}^{\text{task}}(x) = J(q)^{-\top} M(q) J(q)^{-1}$ (pulled back to task space via the Jacobian J), geodesics in task space correspond to straight-line movements in task space, consistent with the empirical observation that reaching movements approximately follow straight trajectories in Cartesian space.

0.150.3 25c.3 Enactivism and Mathematical Limits of Representation

Enactivism (Varela, Thompson, Rosch 1991; Noë 2004) argues that cognition is not the manipulation of internal representations but the *enactment* of sensorimotor contingencies: the know-how of how sensory arrays change in response to action. This is a radical departure from the representationalist paradigm that underlies all of Section s I–XI.

The mathematical challenge for enactivism is to give a precise account of what "sensorimotor know-how" is, without invoking the concept of a stored representation. The dynamical systems framework of Section 25 provides a partial answer: sensorimotor contingencies are **invariants** of the agent-environment coupling dynamics, not stored representations.

Formally, a sensorimotor contingency Γ is a section of the tangent bundle of the sensory manifold $\mathcal{S}$: $\Gamma : \mathcal{A} \times \mathcal{E} \to T\mathcal{S}$, mapping agent-environment configurations to tangent vectors in sensory space. The "knowledge" of Γ is the agent's capability to predict sensory changes from actions, not a stored lookup table but a functional competence embedded in the agent's neural and bodily dynamics.

The **limits** of the enactivist framework are mathematical as much as philosophical:

1. Sensorimotor contingency theory lacks a principled account of object permanence — the representation of objects when they are not currently perceived. A purely dynamical account must explain how an agent maintains "knowledge" of occluded objects without storing a representation; the standard answer (extended through the body and environment) is suggestive but not yet mathematically precise.

2. The theory of affordances (Gibson 1979) — the action possibilities offered by the environment — requires a formal definition of what it means for a given agent to "afford" an action. Information-theoretic approaches (affordances as regions of high mutual information between agent actions and environmental changes) or dynamical systems approaches (affordances as the basin of attraction of a motor program) exist but are not yet integrated.

3. The relationship between the continuous dynamics of enactivism and the discrete symbols of language and reasoning requires a mathematical bridge. Neural field theories and attractor network models provide fragments of this bridge but not a complete theory.

0.151 25. Dynamical Systems: The Mathematics of Coupling

0.151.1 25.1 Continuous Dynamical Systems

A continuous-time dynamical system is a triple $(\mathcal{M}, \phi_t, t \in \mathbb{R})$ where $\mathcal{M}$ is a **state space** (a manifold) and $\phi_t : \mathcal{M} \to \mathcal{M}$ is a one-parameter group of smooth maps (**flow**) satisfying $\phi_0 = \text{id}$, $\phi_{t+s} = \phi_t \circ \phi_s$. The flow is generated by a **vector field** $f : \mathcal{M} \to T\mathcal{M}$ (a section of the tangent bundle) via the differential equation $\dot{x}(t) = f(x(t))$.

The fundamental objects of analysis are the system's **invariant sets**: sets $A \subseteq \mathcal{M}$ with $\phi_t(A) = A$ for all t. Fixed points ($f(x^*) = 0$), periodic orbits, and chaotic attractors are invariant sets of qualitatively different character. The **attractor** A of the system is the minimal invariant set with an open basin of attraction: an open neighbourhood $U \supset A$ such that $\lim_{t\to\infty} d(\phi_t(x), A) = 0$ for all $x \in U$.

The **stability** of a fixed point x^* is characterised by the eigenvalues of the Jacobian $Df(x^*)$: if all eigenvalues have negative real parts, x^* is a stable node or spiral (an **attractor**); if any eigenvalue has positive real part, x^* is unstable.

0.151.2 25.2 Agent-Environment Coupling

The embodied cognition framework models a cognitive agent and its environment as a coupled dynamical system. Let $\mathbf{a}(t) \in \mathcal{A}$ be the agent's state (neural and bodily) and $\mathbf{e}(t) \in \mathcal{E}$ be the environment's state. The coupled system is:

$$\dot{\mathbf{a}} = f_A(\mathbf{a}, \mathbf{e}),$$

$$\dot{\mathbf{e}} = f_E(\mathbf{e}, \mathbf{a}).$$

The agent and environment are not independent: each is an input to the other's dynamics. The agent cannot be understood in isolation, and neither can the environment. Cognition, on this view, is a property of the coupled system, not of the agent alone.

The **sensorimotor loop** is captured by the coupling: the agent's effectors act on the environment (the $\mathbf{a}$-dependence of $\dot{\mathbf{e}}$) and the environment's state affects the agent's sensory inputs (the $\mathbf{e}$-dependence of $\dot{\mathbf{a}}$). The loop runs in real time, with no clean separation between perception and action.

Sensorimotor contingencies (O'Regan and Noë) are the mathematical invariants of this loop: the regular relationships between the agent's actions and the resulting sensory changes. Formally, a sensorimotor contingency is a function $\Gamma : \mathcal{A} \times \mathcal{E} \to T\mathcal{S}$ (where $\mathcal{S}$ is the sensory space) mapping agent-environment configurations to the tangent vectors on the sensory manifold generated by small actions. Perception, on this account, is the implicit knowledge of sensorimotor contingencies — knowledge of how sensory arrays change in response to action.

0.152 26. Mathematical Limits: What Formal Cognition Cannot Capture

0.152.1 26.1 The Grounding Problem

A formal symbol system — a Bayesian network, a probabilistic program, a Montague grammar, an IIT circuit — manipulates symbols according to formal rules. But symbols have meaning only in virtue

of their connection to what they represent. This **grounding problem** (Harnad) asks how symbols can acquire meaning, and whether a purely formal system can ever be genuinely semantically grounded.

The mathematical version of this question is sharpened by a result that echoes Quine's indeterminacy in Section XII of the companion series. Let F be any computable function from sensory inputs to symbolic outputs (a model, a neural network, a probabilistic program). Then for any interpretation function $\mathcal{I}$ consistent with F's input-output behaviour, there are uncountably many interpretations $\mathcal{I}'$ also consistent with that behaviour but assigning different meanings to at least one symbol.

Proof sketch. Let $D = \{(x_i, F(x_i))\}$ be the observable input-output pairs of F. For any bijection σ on the symbol domain that preserves the structure of F's outputs (a "permutation symmetry"), the reinterpreted model $F' = \sigma \circ F$ has the same input-output behaviour as F but assigns σ-permuted meanings. The set of such permutations is (generically) uncountably infinite. QED.

Embodied grounding is the proposed resolution: meaning is not in the symbols themselves but in the agent's sensorimotor engagements with the world. The symbol "red" has its meaning in virtue of the sensorimotor contingencies associated with red objects — how they appear when viewed from different angles, how they change in lighting, how to act to maintain or alter that appearance. This is not a formal property but a dynamic one, and its mathematical formalisation lives in the theory of coupled dynamical systems, not in probability distributions over symbolic states.

0.152.2 26.2 The Limits of the Bayesian Framework

We have developed the Bayesian framework over eleven Section s as the central formal tool for modelling cognition. It is now time to assess its limits with the same rigour we applied to language models in the companion series.

Limit 1: Prior specification. Bayesian inference requires a prior $p(\theta)$. In hierarchical models, priors are themselves inferred from hyperpriors. But at some level, priors must be specified by the modeller, and different specifications yield radically different posteriors. The **prior sensitivity problem** is that posterior conclusions can depend more on prior specification than on the data, particularly in low-data regimes. No formal method exists for specifying a prior that is genuinely non-informative in all parameterisations simultaneously (Jeffreys priors are non-informative in one sense but not all).

Limit 2: Model misspecification. Bayesian inference is optimal within a specified model class. But if the true data-generating process is not in the model class, the posterior concentrates on the "closest" model in the KL sense — which may be qualitatively wrong. The Bayesian framework has no mechanism for detecting or recovering from radical model misspecification; it is inherently conservative, committed to the hypothesis class specified in advance.

Limit 3: Computational intractability. As noted in Section II, exact Bayesian inference is intractable for all but the simplest models. Approximate inference methods introduce approximation errors whose effect on downstream conclusions is generally difficult to quantify. The brain's approximate inference must therefore be judged not against the ideal Bayesian optimum but against the best achievable under biologically realistic computational constraints.

Limit 4: The problem of consciousness. IIT aside, the Bayesian framework is silent on consciousness. A Bayesian inference machine performs computations; it makes predictions; it updates beliefs. Whether it experiences anything is a question the mathematics cannot answer. This is not a gap to be filled by future

developments within the framework; it is, as Chalmers argues, a conceptual gap. The mathematical study of cognition must be complemented by a mathematical theory of consciousness — which IIT attempts, with all the difficulties surveyed in Section V.

Limit 5: The systematicity argument. Fodor and Pylyshyn argued that connectionist (and probabilistic) models cannot account for the systematicity of cognition: that any system that can think "John loves Mary" can also think "Mary loves John." The categorical approach of Section XI provides a partial response: compositionality by functoriality guarantees systematicity. But whether neural networks can implement genuine functors, rather than approximations to them, remains an open question linking the mathematics of category theory to the empirics of neural computation.

0.153 26b. The Mathematics of Semantic Spaces

0.153.1 26b.1 Gärdenfors' Conceptual Spaces

Conceptual space theory (Gärdenfors 2000, 2014) proposes that cognitive representations have a geometric structure: concepts are convex regions in a multi-dimensional **quality space**, where dimensions correspond to psychologically salient properties (colour, temperature, size, shape, ...).

Formally, a conceptual space is a metric space $(\mathcal{C}, d)$ together with a set of **quality dimensions** $Q_1, \ldots, Q_n$: measurable functions $Q_i : \mathcal{C} \to \mathbb{R}^{d_i}$ such that $\mathcal{C} \cong \prod_i \mathbb{R}^{d_i}$ as a topological space (the space decomposes as a product of quality dimensions). The metric d is typically a weighted Euclidean metric: $d(x, y) = \sqrt{\sum_i w_i \|Q_i(x) - Q_i(y)\|^2}$, with weights w_i encoding the relative salience of each dimension.

A **concept** is a region $\mathcal{R} \subseteq \mathcal{C}$, and the theory makes the **convexity hypothesis**: natural concepts correspond to **convex regions** of the conceptual space. If points x and y are both in concept C, then every intermediate point $(1 - t)x + ty$ (for $t \in [0, 1]$) is also in C. This hypothesis is empirically supported by the typicality gradient: more central (prototypical) members of a concept are judged as better examples, and the judgment follows the geometric distance from the prototype.

Natural properties (predicates expressed by single words in many languages) are hypothesised to be convex in conceptual space. The **Voronoi tessellation** of exemplars gives the optimal partition of conceptual space into convex regions: each region is the set of points closer (in the conceptual distance d) to one exemplar than to any other. This geometric model of categorisation is equivalent to a nearest-prototype classifier in the metric space.

0.153.2 26b.2 The Information Geometry of Conceptual Change

Scientific and developmental concept change can be modelled as **deformation of conceptual spaces**. When a concept changes meaning (the extension of "fish" changes when we learn that whales are mammals), this corresponds to a change in the Voronoi boundaries of the conceptual space — a geometric deformation.

Information geometry (Section 3c) provides a natural framework: the space of probability distributions over conceptual space is a statistical manifold, and concept change is a path through this manifold. Gradual revision (Bayesian updating) follows a geodesic of the Fisher-Rao metric; revolutionary revision (Kuhnian paradigm shift) may follow a discontinuous path.

The **information-geometric distance** between two conceptual frameworks p and q is the Fisher-Rao geodesic distance:

$$d_{FR}(p,q) = 2\arccos\left(\int \sqrt{p(x)q(x)}\,dx\right) = 2\arccos(\mathrm{BC}(p,q)),$$

where $\mathrm{BC}(p,q) = \int \sqrt{p(x)q(x)}\,dx$ is the **Bhattacharyya coefficient**. This is the geodesic distance on the L^2 sphere of square-root densities, known as the **Hellinger distance** arc: $d_{FR} = 2\arccos(1 - \frac{1}{2}\int(\sqrt{p} - \sqrt{q})^2)$.

A Bayesian learner updates along the m-geodesic (mixture connection) of the statistical manifold; a rational concept change that minimises information loss while incorporating new constraints follows the e-geodesic (exponential connection). These two geodesics are generally different, explaining why Bayesian updating and conceptual revision may feel very different from the learner's perspective despite being formally related.

0.153.3 26b.3 Persistence Homology and the Topology of Neural Representations

Topological data analysis (TDA; Carlsson 2009) provides tools for detecting the global topological structure of data without assuming a specific geometric model. The central tool is **persistent homology**: it detects loops, voids, and higher-dimensional "holes" in data at all scales simultaneously.

For a point cloud $X = \{x_1, \ldots, x_n\} \subset \mathbb{R}^d$ (e.g., neural population activity vectors), the **Vietoris-Rips filtration** builds a simplicial complex $\mathcal{R}_\epsilon$ by:

- Adding a vertex for each point x_i.
- Adding an edge between x_i and x_j if $\|x_i - x_j\| \leq \epsilon$.
- Adding a k-simplex $[x_{i_0}, \ldots, x_{i_k}]$ if all pairwise distances are $\leq \epsilon$.

As ϵ increases from 0 to ∞, topological features appear (are "born") and disappear ("die"). The **persistence diagram** records the birth-death pairs (ϵ_b, ϵ_d) for each topological feature (connected components in dimension 0, loops in dimension 1, voids in dimension 2, ...). Features with large persistence $\epsilon_d - \epsilon_b$ are considered topologically significant; features with small persistence are noise.

For neural data (Gardner et al., 2022), persistent homology has revealed that the hippocampal place cell population encodes location as a **torus** (T^2): the persistent homology shows two independent loops (generators of $H_1(T^2;\mathbb{Z}) \cong \mathbb{Z}^2$) at the scale corresponding to the room's dimensions, exactly the topological signature of a torus. This provides mathematical evidence that the hippocampus uses toroidal geometry to represent two-dimensional space — a prediction of grid cell theory (the hexagonal grid is a quotient of $\mathbb{R}^2$ by a lattice, and its universal cover has the topology of a torus).

0.154 26h. Embodied Intelligence and Mathematical Frontiers

0.154.1 26h.1 Morphological Computation

Morphological computation (Pfeifer and Bongard 2007; Hauser et al., 2011) is the idea that the body's physical structure performs computation, offloading work from the neural system. The compliance, in-

ertia, and morphology of the body solve control problems that would otherwise require explicit neural computation.

Passive dynamic walking (McGeer 1990) demonstrates morphological computation in locomotion: a bipedal mechanism with no motors and no control signals can walk stably down a shallow slope, powered only by gravity. The mechanical structure (leg geometry, foot shape, mass distribution) implements the stable limit cycle of walking without any active computation. The information-theoretic formulation: the body's mechanical dynamics compute the walking trajectory, reducing the neural information required to $I_{\text{control}} = I_{\text{neural}} - I_{\text{morphological}}$ bits (subtracting the information provided by morphological dynamics).

Soft robotics and computation in compliance. Compliant (soft) bodies naturally implement low-pass filtering of motor commands (compliance absorbs high-frequency noise), nonlinear dynamics that perform computation (a soft manipulator arm naturally computes the grasp force distribution), and energy storage (compliant tendons store and release elastic energy during locomotion). The **morphological information bound**: for a task requiring I_{task} bits of information, a morphologically optimised body can reduce the neural information requirement to $I_{\text{neural}} \leq I_{\text{task}} - I_{\text{morphology}}$.

0.154.2 26h.2 The Computational Limits of Learning

PAC learning theory (Section 3k) establishes complexity bounds for learning specific hypothesis classes. The most fundamental limit is the **No Free Lunch theorem** (Wolpert 1992; Mitchell 1980):

Theorem 0.154.1 (No Free Lunch for Learning)**.** For any learning algorithm A and any target function f in the set of all possible functions $\mathcal{F}$, the average generalisation error of A on f, averaged uniformly over all $f \in \mathcal{F}$, equals the generalisation error of random guessing.

The theorem says: without prior assumptions (inductive bias), no learning algorithm can generalise. Any algorithm that outperforms random guessing on some functions must perform worse than random guessing on others. The theorem forces us to take inductive bias seriously: the choice of hypothesis class, kernel, or prior is not arbitrary — it determines what can be learned.

For cognitive science, the No Free Lunch theorem establishes that human intelligence must rely on strong inductive biases (built-in or learned from early experience) to generalise rapidly from few examples. The question "How does the brain learn?" must be answered jointly with "What are the brain's inductive biases?" The Bayesian framework provides a principled account of inductive bias as the prior distribution: the prior encodes what the learner assumes about the world before any data are seen.

0.155 26g. Phase Transitions and the Emergence of Consciousness

0.155.1 26g.1 First-Order Transitions and Conscious Access

The global workspace theory (Section 20c) predicts that conscious access involves a sudden, all-or-nothing transition: either the workspace ignites (and the stimulus is consciously perceived) or it does not. This is characteristic of a **first-order phase transition**: the order parameter (global workspace activity) jumps discontinuously at the critical input strength.

Mathematically, for a neural mass model of the workspace, the steady-state equations have two solutions for intermediate input strengths: a low-activity (unconscious) state and a high-activity (conscious)

state. The transition between them occurs at a saddle-node bifurcation (Section 25), which is a first-order transition in the thermodynamic sense.

Evidence. Psychophysical experiments show that the detection probability for near-threshold stimuli is often bimodal (Sergent and Dehaene 2004): subjects either report clear conscious perception (high confidence) or complete absence of perception (low confidence), with few intermediate reports. This bimodality is consistent with a first-order transition but inconsistent with a gradual increase in evidence strength (which would predict a unimodal, graded distribution of confidence). The **pre-stimulus alpha power** predicts which regime the brain is in: high alpha (10 Hz oscillations) in parietal cortex → low-activity state → subliminal; low alpha → high-activity state → supraliminal.

0.155.2 26g.2 Critical Brain Dynamics and Consciousness

The **critical brain hypothesis** (Shew and Plenz 2013; Beggs and Timme 2012) proposes that the brain operates near a critical point, exhibiting scale-free dynamics (power-law distributions of avalanche sizes and durations). Critical dynamics are hypothesised to be necessary for consciousness because:

1. **Maximal dynamic range**: near the critical point, the brain can respond to stimuli over many orders of magnitude (the "1/f" noise structure provides optimal sensitivity across timescales).
2. **Maximal information transmission**: critical dynamics maximise the transfer entropy and directed information between brain regions (Section 13e).
3. **Maximal integration**: near the critical point, the Fisher information matrix of the neural state is maximised (the inverse is the susceptibility, which diverges at the critical point). This maximal Fisher information corresponds to maximum discriminability between nearby states — a prerequisite for the high information content of conscious experience.

Mathematical description. The Wilson-Cowan field equations (Section 26c) have a critical point at $J_{\text{eff}} = 1$ (the effective gain equals 1): at this point, small perturbations neither grow (subcritical) nor decay (supercritical) but persist and propagate across the field. The distribution of spatiotemporal avalanche sizes is a power law $P(\text{size}) \propto \text{size}^{-3/2}$ (Landau-Ginzburg mean-field exponent), consistent with experimental observations in cortical slices and in vivo recordings.

0.156 26f. Integrated Mathematical Perspectives on Cognition

0.156.1 26f.1 The Bayesian Brain as a Unified Framework

Across the twelve Section s of this monograph, a single mathematical framework has repeatedly emerged as the unifying principle: **Bayesian inference with precision-weighted prediction errors**. Whether we are studying perception (Section IV), language (Section XI), reasoning (Section VII), or motor control (Section X), the same mathematical structure appears:

- A **generative model** $p(\mathbf{x}, \mathbf{z})$ encoding beliefs about how observations $\mathbf{x}$ are generated from hidden causes $\mathbf{z}$.
- A **recognition model** $q(\mathbf{z}|\mathbf{x})$ approximating the posterior over causes.
- A **free energy** $\mathcal{F}(q) = \text{KL}(q||p) - \log p(\mathbf{x})$ that is jointly minimised by updating both the recognition model and (through action) the observations.

The precision matrices Π (inverse covariances) play the role of **attention**: they determine how much weight to assign to prediction errors from different sources. High precision $\rightarrow$ strong sensory evidence; low precision $\rightarrow$ weak, uncertain evidence. The brain's attentional systems modulate these precision weights, implementing a hierarchical inference architecture that integrates information across multiple spatial and temporal scales.

0.156.2 26f.2 Information Geometry as the Meta-Framework

Behind the Bayesian framework lies **information geometry**: the study of the intrinsic Riemannian structure of the space of probability distributions. The Fisher-Rao metric (Section I) is the natural metric on this space; the α-connections describe the different ways information can "flow" through the space; the natural gradient (Section 3i) provides the geometrically correct direction for optimisation.

Every major concept in this monograph has an information-geometric interpretation:

- **Bayesian updating** is parallel transport along an m-geodesic.
- **Variational inference** is m-projection onto a restricted manifold.
- **Active Inference** is gradient descent on the variational free energy manifold.
- **IIT's** Φ is the Wasserstein distance between two points on the manifold of cause-effect structures.
- **Kolmogorov complexity** is the information content of a point on the universal statistical manifold.

0.156.3 26f.3 Open Mathematical Problems in Mathematical Cognitive Science

We conclude with a list of open mathematical problems at the frontier of the field:

1. **The complexity of Φ with approximations.** Does there exist a polynomial-time approximation algorithm for Φ with provable approximation ratio? The Wasserstein-1 computation can be approximated via entropy-regularisation (Sinkhorn algorithm) in $O(n^2/\epsilon^2)$ time; can the full Φ be computed with similar guarantees?
2. **Identifiability in Active Inference.** Under what conditions is the generative model of an Active Inference agent identifiable from its behaviour? For classical POMDPs, the optimal policy does not uniquely identify the reward function (inverse reinforcement learning is ill-posed without additional priors). Does AIF's additional epistemic component improve identifiability?
3. **Convergence of predictive coding in hierarchies of arbitrary depth.** The predictive coding dynamics are gradient descent on the free energy; under what conditions on the generative functions $g^{(l)}$ does this converge to the true posterior? For linear $g^{(l)}$, convergence is guaranteed by the positive definiteness of the Hessian. For general nonlinear $g^{(l)}$, convergence conditions are unknown.
4. **The mathematical foundations of embodied cognitive science.** Can the dynamical systems approach (Section 25) and the representationalist Bayesian approach (Section s I–XI) be unified in a single mathematical framework? The **enactive Bayesian brain** hypothesis (Clark 2019; Pezzulo et al., 2016) proposes that active inference provides this unification: representations are approximations to the generative model's internal states, and their dynamical character is the continuous free energy minimisation process. A rigorous mathematical formulation of this view remains an open problem.

5. **Information-geometric characterisation of consciousness.** Can the axiomatic framework of IIT (Section 11) be derived as a theorem from information geometry, rather than postulated? Specifically: is Φ the unique information-geometric invariant that satisfies the five IIT postulates? A negative answer would open the door to alternative measures of consciousness with better computational properties.

These problems sit at the intersection of mathematics, theoretical neuroscience, and cognitive science. Their resolution will require new mathematical tools that do not yet exist, and their solutions will deepen our understanding of the most fundamental questions about mind, intelligence, and consciousness.

0.157 26e. Self-Organisation, Emergence, and Complexity in Cognition

0.157.1 26e.1 Complex Adaptive Systems and the Edge of Chaos

Complex adaptive systems (CAS; Holland 1995; Kauffman 1993) are systems of many interacting agents that adapt to their environment and to each other, producing emergent global behaviours not predictable from individual components. The brain is the prototypical CAS: billions of neurons interact locally, producing global cognitive functions that are not properties of any individual neuron.

The **edge of chaos** hypothesis (Langton 1990; Kauffman 1993): complex systems operating near a **phase transition** between ordered (periodic, predictable) and disordered (chaotic, unpredictable) dynamics exhibit the most complex, adaptive behaviour. Near the critical point, the system has the longest-range correlations, the largest dynamic range of responses, and the greatest information transmission.

For random Boolean networks (Kauffman models), the critical connectivity is $K_c = 2$ (each node connected to exactly 2 others on average). Below K_c: frozen, ordered dynamics. Above K_c: chaotic dynamics. At K_c: critical dynamics with power-law avalanche distributions. Empirically, neural activity in cortex shows power-law distributions of avalanche sizes and durations (Beggs and Plenz 2003), consistent with critical dynamics.

Directed information and criticality. At the critical point, the **dynamic susceptibility** $\chi = \partial\langle m\rangle/\partial h$ (response of the order parameter to a perturbation) diverges: the system is maximally sensitive to inputs. In information-theoretic terms, the **transfer entropy** $T_{X\to Y} = \mathrm{I}(Y_t; X_{t-1}|Y_{t-1})$ (directed information from X to Y) is maximised at criticality: the system transmits the most information about future states given past states when operating at the critical point. This provides an information-theoretic justification for the edge-of-chaos hypothesis: critical dynamics optimise information transmission.

0.157.2 26e.2 Renormalisation Group and Multi-Scale Cognition

The **renormalisation group** (RG; Wilson 1971) is a mathematical framework for understanding how systems look different at different scales. Starting from a microscopic description (individual neurons), RG coarse-grains the system by averaging over small-scale fluctuations and asking what effective description emerges at each scale.

For a neural field with coupling constant J (the strength of excitatory connections) and external input h, the RG flow describes how (J, h) change as we average over microscopic fluctuations:

$$\frac{dJ}{d\ell} = (2-\eta)J - \lambda J^3 + O(J^5), \qquad \frac{dh}{d\ell} = (d - y_h)h,$$

where ℓ is the logarithmic scale factor, η is the anomalous dimension, λ is a coupling constant, and y_h is the scaling dimension of the field. The fixed points of the RG flow (where $dJ/d\ell = 0$) correspond to scale-invariant states: the **Gaussian fixed point** ($J = 0$: non-interacting) and the **Wilson-Fisher fixed point** ($J^2 = (2 - \eta)/\lambda$: interacting, corresponding to critical dynamics).

For multi-scale cognition, the RG perspective suggests that cognitive representations at different levels of abstraction (sensory features, objects, categories, concepts) correspond to coarse-grained descriptions of the same underlying system at different scales. Learning is the process of finding the RG flow that maps from sensory-level descriptions to high-level conceptual descriptions: each layer of a deep network performs one step of the RG coarse-graining.

0.157.3 26e.3 Integrated Information Geometry and Consciousness

Connecting the themes of Section s V and XII: consciousness (integrated information Φ) has a geometric interpretation in terms of the information geometry of the system's state space.

The cause-effect structure of a system is a manifold in the space of probability distributions: the set of all cause-effect repertoires $\{p_{\text{CE}}(m, M) : m \in M, M \subseteq \text{System}\}$ forms a family of distributions parametrised by the mechanism and state. The geometry of this manifold (curvature, dimension, geodesic distances) captures the system's integrated information structure.

Φ is the Wasserstein distance between two points on this manifold: the intact CES and the MIP CES. In information geometry terms, Φ is the distance between the system's actual cause-effect geometry and the most decomposable approximation consistent with the partition. Systems with high Φ have cause-effect geometries that are far from any product manifold: their integrated structure cannot be approximated by independent parts.

The geometric perspective suggests a **differential geometry of consciousness**: the gradient of Φ with respect to the system's parameters (connection strengths, time constants) describes how changes in neural architecture affect conscious experience. Maximising Φ along a gradient ascent corresponds to optimising the neural architecture for maximal integration — a developmental or evolutionary pressure that could, in principle, explain the evolution of complex nervous systems with high Φ.

0.158 26d. Quantum Cognition: A Mathematical Assessment

0.158.1 26d.1 The Quantum Probability Framework

Quantum cognition (Busemeyer and Bruza 2012; Pothos and Busemeyer 2013) proposes that the formal mathematics of quantum mechanics (Hilbert spaces, projection operators, Born rule) provides a better description of human judgment and decision-making than classical probability theory. The claim is not that the brain is a quantum computer, but that quantum probability theory is a more accurate mathematical model of cognition.

The quantum probability framework replaces the classical sample space Ω (a set of mutually exclusive events) with a Hilbert space $\mathcal{H}$ and represents:

- **Cognitive states**: unit vectors $|\psi\rangle \in \mathcal{H}$ (or more generally, density matrices ρ).
- **Events**: closed subspaces $\mathcal{S} \subseteq \mathcal{H}$, represented by projection operators $P_{\mathcal{S}}$.

- **Probabilities**: $P(A|\psi) = \langle\psi|P_A|\psi\rangle$ (Born rule).
- **Sequential events**: $P(A \text{ then } B|\psi) = \|P_B P_A|\psi\rangle\|^2$ (sequential projection).

The key difference from classical probability: for non-commuting projections ($[P_A, P_B] \neq 0$), $P(A \text{ then } B) \neq P(B \text{ then } A)$. This **non-commutativity** captures **order effects** in judgment: the probability of a response depends on the order in which questions are asked.

0.158.2 26d.2 The Conjunction Fallacy and Quantum Interference

The **conjunction fallacy** ($P(\text{A and B}) > P(\text{B})$ in human judgments) is a violation of classical probability theory. In quantum probability, this is not a violation but a consequence of interference:

$$P_q(A \wedge B) = \|P_B P_A|\psi\rangle\|^2 = P_c(A \wedge B) + 2\text{Re}(\langle\psi|P_{A\setminus B}P_{A\cap B}|\psi\rangle),$$

where the second term is the **interference term**. When this term is positive (constructive interference), $P_q(A \wedge B) > P_c(A \wedge B)$, and the conjunction fallacy can arise without violating any internal consistency requirement.

The quantum model makes quantitative predictions: the interference term should be correlated across individuals and should have a specific signature in the biconditional probability $P(A|B) \cdot P(B) = P(B|A) \cdot P(A) +$ interference. Fitting quantum models to behavioral data allows estimation of the state vector $|\psi\rangle$ and projection operators P_A, P_B, and the predicted interference term.

0.158.3 26d.3 Limitations and Assessment

The quantum cognition program has been criticised on several grounds:

Model comparison: quantum models have more free parameters than classical models (the state vector $|\psi\rangle$ has $2d - 2$ real degrees of freedom for a d-dimensional Hilbert space). Without careful model comparison (AIC, BIC, or Bayes factors), apparent superiority of quantum models may reflect overfitting.

Physical vs. mathematical interpretation: using quantum formalism as a mathematical convenience (not implying physical quantum effects) is legitimate but requires justification for why quantum rather than any other non-classical probability theory.

Alternative explanations: many phenomena attributed to quantum interference have explanations in classical probability theory with appropriate sampling models (like the sampling model of the conjunction fallacy: Linda is representative of the category "feminist bank tellers" and people sample from this category to estimate probability). Distinguishing quantum and classical explanations requires careful experimental design.

The mathematical contribution of quantum cognition is real and significant: it demonstrates that Birkhoff-von Neumann quantum logic (Section 16d) provides a formally consistent non-classical probability theory that captures human judgment patterns. Whether this is the correct description of cognition or a useful approximation remains an open empirical question.

0.159 26c. Neural Field Theory: Mathematics of Brain Dynamics

0.159.1 26c.1 Wilson-Cowan Neural Field Equations

Neural field theory (Wilson and Cowan 1972, 1973; Amari 1977) models the dynamics of large-scale cortical activity as a continuous field $u(x,t)$ (the mean membrane potential at cortical position x at time t), governed by integro-differential equations:

$$\tau \frac{\partial u(x,t)}{\partial t} = -u(x,t) + \int_{\Omega} w(x,x') f(u(x',t))\, dx' + I(x,t),$$

where τ is the membrane time constant, $w(x,x')$ is the **synaptic weight kernel** (the strength and sign of the connection from position x' to x), $f(\cdot)$ is the **firing rate function** (typically a sigmoid $f(u) = 1/(1+e^{-\beta(u-\theta)})$), and $I(x,t)$ is external input.

The weight kernel $w(x,x')$ in primary visual cortex has a characteristic **Mexican hat profile**: excitatory connections at short range ($|x-x'| < \sigma_e$) and inhibitory connections at medium range ($\sigma_e < |x-x'| < \sigma_i$), with both decaying at longer ranges. This "excitation-inhibition" lateral connectivity implements a **spatial low-pass filter** (amplifies broad activity patterns) combined with **lateral inhibition** (suppresses nearby activity), producing contrast enhancement and edge detection.

Pattern formation (Turing instabilities). The homogeneous steady state $u^* = f(u^*)(\int w)$ loses stability when the Fourier transform $\hat{w}(k)$ of the kernel becomes positive for some spatial frequency k^*. At this **Turing bifurcation**, spontaneous spatially periodic patterns form with wavelength $2\pi/k^*$. For the Mexican hat kernel, k^* corresponds to the characteristic spacing of cortical columns (approximately 0.8–1 mm in visual cortex), predicting the columnar organisation of visual cortex from purely dynamical principles.

0.159.2 26c.2 Attractor Dynamics and Working Memory

Working memory in prefrontal cortex has been modelled as a **bump attractor**: a localised bump of activity that forms and persists in the neural field, representing the memorised item's location or value.

For a one-dimensional ring model (representing a circular feature like angle), the attractor states are solutions of:

$$0 = -u(x) + \int_0^{2\pi} w(x-x') f(u(x'))\, dx',$$

i.e., steady-state solutions of the field equation. For a weight kernel $w(\Delta x) = J_{\text{exc}} \exp(-\Delta x^2/2\sigma_e^2) - J_{\text{inh}}$ (Gaussian excitation, global inhibition), bump solutions exist for $J_{\text{exc}} > J_{\text{inh}}(2\pi\sigma_e^2)^{1/2}$. The bump can be centred at any angle $\phi \in [0, 2\pi)$: this continuous family of attractors is the mathematical substrate of **continuous attractor neural networks**, which model the representation of continuous variables (angle, position, time) in neural activity.

Stability analysis. The bump at angle ϕ is stable with respect to perturbations in all directions except the **zero mode** (uniform shift of the bump), which has zero eigenvalue (Goldstone mode). This zero mode corresponds to the fact that the bump can slide to any position without energy cost — the attractor has a continuous symmetry (translation invariance) that generates a family of marginal stabilities. In biology, noise slowly diffuses the bump along this zero mode, accounting for the observed drift and variance in working memory representations over delay intervals.

0.159.3 26c.3 Oscillations, Synchrony, and Binding by Synchrony

The **binding by synchrony** hypothesis (Singer and Gray 1989; Engel and Singer 2001) proposes that the binding of distributed neural representations (features of the same object) is achieved by **synchronous oscillations** in the gamma band (30–80 Hz). Features of the same object are represented by neurons that fire in synchrony; features of different objects are represented by neurons in different phase relationships.

Mathematically, the synchrony of two oscillating neurons can be modelled by coupled phase oscillators (Kuramoto model):

$$\frac{d\phi_i}{dt} = \omega_i + \sum_j K_{ij} \sin(\phi_j - \phi_i),$$

where ϕ_i is the phase of neuron i, ω_i is its natural frequency, and K_{ij} is the coupling strength. The **Kuramoto order parameter** $r = |\frac{1}{N}\sum_j e^{i\phi_j}|$ measures synchrony: $r = 1$ for perfect synchrony, $r \approx 0$ for incoherence.

The critical coupling strength for the onset of synchrony (Kuramoto 1984):

$$K_c = \frac{2}{\pi g(\omega^*)},$$

where $g(\omega)$ is the frequency distribution (assumed unimodal, symmetric about ω^*). For $K > K_c$, a macroscopically coherent state develops: a fraction of neurons synchronise, producing a non-zero r. This **synchronisation transition** is a second-order phase transition.

For binding by synchrony to work, the coupling K_{ij} between neurons representing features of the same object must be positive (synchronising), while coupling between neurons representing features of different objects must be negative (anti-synchronising) or zero. The **binding field equation** for the coupling:

$$K_{ij} = A s_{ij} - B(1 - s_{ij}),$$

where $s_{ij} = 1$ if i and j represent features of the same object, $s_{ij} = 0$ otherwise. The task of the binding circuit is to determine s_{ij} from the input, which is equivalent to image segmentation — a computationally hard problem. The Kuramoto model of binding thus re-formulates segmentation as an optimisation problem: find the phase assignment that minimises the coupling energy $E = -\sum_{ij} K_{ij} \cos(\phi_i - \phi_j)$.

0.160 27. Open Frontiers: Theorems at the Edge of Understanding

We close with five mathematical problems that represent the open frontier of the mathematics of cognition.

Problem 1: The Neural Code. What is the mathematical structure of the neural code — the mapping from external stimuli and internal states to neural firing patterns? Population vector codes, sparse codes, rate codes, and temporal codes have all been proposed. The problem is to identify the mathematical invariant (the "sufficient statistic" of neural activity for behaviour) and characterise its information-geometric properties. Partial progress: information-theoretic lower bounds on the neural code's efficiency relative to the efficient coding hypothesis (Barlow).

Problem 2: Consciousness and Causation. IIT identifies consciousness with integrated information. The free-energy principle identifies perception with prediction-error minimisation. Are these two frameworks compatible? Can a unified mathematical framework capture both the intrinsic causal structure of

conscious experience and the extrinsic predictive dynamics of the Bayesian brain? A precise formulation of this question requires the tools of both information geometry (for IIT) and variational calculus (for the free-energy principle).

Problem 3: Learning Causal Structure. The do-calculus of Section VII tells us how to compute causal effects once we know the causal graph. But how should a cognitive agent learn the causal graph from observational data? The constraint-based methods (Peter-Clark algorithm) and score-based methods (GES algorithm) provide partial answers for linear Gaussian systems. For non-linear systems and discrete variables in small samples — the setting of most cognitive development — the theory is incomplete.

Problem 4: The Compositionality-Flexibility Trade-off. Categorical compositionality guarantees systematic generalisation. Neural networks trained on data exhibit flexible generalisation that often outperforms symbolic systems on naturalistic tasks but fails on structured tests like SCAN. The mathematical problem is to characterise the precise relationship between the degree of compositionality (measured categorically, e.g., by the degree to which the neural network approximates a functor) and generalisation performance across different task distributions.

Problem 5: Embodied Grounding and Symbol Emergence. How do grounded symbols emerge from sensorimotor dynamics? The dynamical systems perspective of this Section provides a framework (sensorimotor contingencies, attractors, coupling) but not yet a complete theory of how discrete symbolic representations arise from continuous dynamical processes. The mathematical challenge is to construct a formal bridge between the geometry of sensorimotor manifolds and the combinatorial structure of symbolic cognition — a question that lies at the intersection of differential geometry, category theory, information theory, and dynamical systems, and whose resolution may require mathematical tools that do not yet exist.

0.161 3p. Robust Bayesian Estimation and M-Estimators

0.161.1 3p.1 M-Estimators as Approximate MAP

M-estimators (Huber 1964, 1981) are a class of estimators defined as the solution to $\hat{\theta} = \arg\min_\theta \sum_i \rho(x_i, \theta)$ for a **loss function** ρ. Different choices of ρ give different estimators:

- $\rho(x, \theta) = (x - \theta)^2$: least squares (MLE for Gaussian, sensitive to outliers).
- $\rho(x, \theta) = |x - \theta|$: least absolute deviations (robust to outliers, corresponds to Laplace likelihood).
- $\rho(x, \theta) = \text{Huber}(x - \theta; k)$: quadratic for $|x - \theta| \leq k$, linear for $|x - \theta| > k$ (compromise between efficiency and robustness).

The Bayesian interpretation: minimising $\sum_i \rho(x_i, \theta)$ is equivalent to MAP estimation with log-prior $\propto -\Omega(\theta)$ and log-likelihood $\propto -\sum_i \ell(x_i, \theta)$. Robust M-estimators correspond to heavy-tailed likelihood models (Laplace: double-exponential tails, Student-t: polynomial tails) that downweight the influence of outliers on the estimate. The **influence function** $\text{IF}(x; T, F) = \lim_{\varepsilon \to 0}[T((1-\varepsilon)F + \varepsilon\delta_x) - T(F)]/\varepsilon$ measures the sensitivity of estimator T to a single observation at x: a bounded influence function implies robustness to outliers.

0.162 26k. The Hard Problem of Consciousness: Mathematical Perspectives

0.162.1 26k.1 Mathematical Approaches to the Explanatory Gap

The **hard problem of consciousness** (Chalmers 1995) asks why there is subjective experience at all — why physical processes give rise to "something it is like" to be in a mental state. This is distinct from the "easy problems" (explaining cognitive functions, neural correlates, and information processing), which are hard scientifically but not philosophically puzzling.

IIT's response. IIT claims to solve the hard problem by asserting that integrated information *is* experience: $\Phi > 0$ is not merely correlated with consciousness but is identical to it. The identity is stated at the mathematical level: the axioms (existence, composition, information, integration, exclusion) characterise experience mathematically, and Φ is the measure that satisfies all axioms. On this view, asking why $\Phi > 0$ implies consciousness is like asking why high kinetic energy implies high temperature: the identity is mathematical, not empirical.

The mathematical gap. Critics (Tononi's critics, Chalmers himself) note that even if Φ perfectly predicts the presence and properties of conscious experience, this does not explain *why* $\Phi > 0$ gives rise to experience. The mathematical structure might be the right description of consciousness without being an explanation. This is the explanatory gap recast in formal terms: a mathematical identity does not provide a causal or constitutive explanation.

Structuralism. A different mathematical approach: consciousness might be constituted by its **relational structure** (the pattern of similarities and differences between experiences) without any further intrinsic properties. The structure of colour space (a 3-dimensional manifold with specific metric properties) constitutes the phenomenal properties of colour experience. Mathematically, this is the thesis that consciousness is a **functor** from physical structure to phenomenal structure: the relational structure of the cause-effect geometry (Φ, the CES) is mapped to the relational structure of phenomenal experience (the qualitative similarities and differences between experiences). The hard problem is dissolved: there is nothing more to experience than its structural properties, which are given by the mathematical theory.

0.163 3q. Bayesian Experimental Psychology: Model Fitting and Identifiability

0.163.1 3q.1 The Identifiability Problem in Cognitive Models

A parametric cognitive model $\mathcal{M} = \{p(D|\theta) : \theta \in \Theta\}$ is **identifiable** if distinct parameter values generate distinct data distributions: $\theta \neq \theta' \Rightarrow p(\cdot|\theta) \neq p(\cdot|\theta')$. Identifiability is a precondition for meaningful parameter estimation: if two parameter vectors are indistinguishable from data, no statistical method can recover them separately.

Formal definition. $\mathcal{M}$ is **globally identifiable** if $p(D|\theta) = p(D|\theta')$ for all D implies $\theta = \theta'$. It is **locally identifiable** at θ_0 if there exists a neighbourhood $U \ni \theta_0$ such that $p(D|\theta) = p(D|\theta_0)$ for all D implies $\theta = \theta_0$ within U. Local identifiability is equivalent to the Fisher information matrix $\mathcal{I}(\theta_0)$ being non-singular.

The rank condition for identifiability. For a model with k parameters observed through m summary statistics (e.g., first m moments), the model is locally identifiable iff the Jacobian matrix $J = \partial(\mathbb{E}[T_1], \ldots, \mathbb{E}[T_m])/\partial(\theta_1, \ldots, \theta_k)$ has rank k. If rank $J < k$, there exist **unidentified directions**: infinitesimal perturbations of θ in the null space of J that leave all observable statistics unchanged.

Example: the SIMPLE model of serial position effects. The SIMPLE model (Brown et al., 2007) predicts free recall probability from a temporal distinctiveness mechanism. It has three parameters: c (temporal compression), s (variability of temporal representations), and t (response threshold). Analysis shows that the model is locally identifiable only when temporal spacing is varied across lists; for fixed temporal spacing, c and s are collinear in the Jacobian (rank $J = 2 < 3$), making the model non-identifiable from standard free recall data alone.

0.163.2 3q.2 Approximate Bayesian Computation

When the likelihood $p(D|\theta)$ is analytically intractable (as for many cognitive models with complex dynamics), but sampling from the model is feasible, **Approximate Bayesian Computation** (ABC; Beaumont et al., 2002; Marjoram et al., 2003) provides a likelihood-free inference procedure.

The simplest ABC algorithm (ABC-rejection):

1. Sample $\theta^* \sim p(\theta)$ (prior).
2. Simulate $D^* \sim p(\cdot|\theta^*)$.
3. Compute summary statistics $S(D^*)$.
4. Accept θ^* if $\|S(D^*) - S(D_{\text{obs}})\| \leq \epsilon$; reject otherwise.
5. Repeat until N samples are accepted.

The accepted samples approximate the **ABC posterior** $p_\epsilon(\theta|D_{\text{obs}}) \propto p(D_{\text{obs}}|\theta)^{(\epsilon)}p(\theta)$, where $p(D|\theta)^{(\epsilon)} = P(\|S(D^*)-S(D)\| \leq \epsilon|\theta)$ is an approximation to the likelihood. As $\epsilon \to 0$, the ABC posterior converges to the true posterior, but acceptance rates become exponentially small.

ABC-SMC. Sequential Monte Carlo ABC (Del Moral et al., 2012) reduces the computational cost by running a sequence of ABC distributions with decreasing tolerance $\epsilon_1 > \epsilon_2 > \cdots > \epsilon_T$. At each step t, a particle population is propagated from $p_{\epsilon_{t-1}}$ to p_{ϵ_t} via MCMC moves, resampling, and tolerance reduction. The final population at ϵ_T approximates the target posterior.

Summary statistic selection. The choice of summary statistics $S(D)$ critically affects the posterior approximation. Ideally, S should be **sufficient** for θ (no loss of information), but sufficient statistics are unavailable for most complex models. Practical choices for cognitive models: mean RT and variance for drift-diffusion models; accuracy and RT quantiles for choice-RT data; learning curves and final accuracy for reinforcement learning models. Information-theoretic criteria (Fearnhead and Prangle 2012) select S to minimise the expected posterior variance, using a preliminary run of ABC to estimate the posterior semi-automatically.

0.163.3 3q.3 Posterior Predictive Model Checking

Posterior predictive checks (PPCs; Gelman et al., 2004) test whether a Bayesian model adequately describes the data by comparing observed statistics to their distribution under the posterior predictive:

$$p(T_{\text{rep}} \geq T_{\text{obs}}|D) = \int P(T(D_{\text{rep}}) \geq T(D_{\text{obs}})|\theta)\, p(\theta|D)\, d\theta,$$

where $T(D)$ is a test statistic and $D_{\text{rep}} \sim p(\cdot|\theta)$ is a replicated dataset. The **posterior predictive p-value** $p_{\text{PPC}} = P(T_{\text{rep}} \geq T_{\text{obs}}|D)$ measures whether the observed test statistic falls within the range predicted by the model. Values near 0 or 1 indicate systematic discrepancies.

For cognitive models, important PPCs include: the shape of RT distributions (quantile-probability plots), the serial position curve (primacy, recency effects), learning curves, and response time correlations across trials. A model that passes PPCs for the statistics it was fitted to but fails for unmodelled statistics reveals specific inadequacies that motivate model extension.

0.164 12c. Fisher Information and the Geometry of Neural Coding

0.164.1 12c.1 Population Codes and Fisher Information

A population of N neurons with tuning curves $f_i(s)$ (mean firing rates as a function of stimulus s) encodes stimulus information. If the spike counts are Poisson-distributed, $r_i \sim \text{Poisson}(f_i(s))$, the Fisher information for s from the full population is:

$$\mathcal{I}_{\text{pop}}(s) = \sum_{i=1}^{N} \frac{(f_i'(s))^2}{f_i(s)},$$

by the additivity of Fisher information for independent observations. The Cramér-Rao bound states that any unbiased estimator $\hat{s}$ satisfies $\text{Var}(\hat{s}) \geq 1/\mathcal{I}_{\text{pop}}(s)$. The population vector decoder achieves this bound asymptotically in the limit of large N and high signal-to-noise ratio.

Optimal tuning curves. For a fixed number of neurons and a fixed total firing rate, what tuning curve shapes maximise the population Fisher information? Ganguli and Simoncelli (2014) showed that for a stimulus with prior $p_S(s)$, the optimal tuning curves are arranged so that the Fisher information tracks the prior:

$$\mathcal{I}_{\text{pop}}(s) \propto p_S(s)^2.$$

This **Fisher information matching** principle predicts that the density of neurons should be highest where stimuli are most common. For orientation in primary visual cortex, stimuli near horizontal and vertical are more common in natural scenes, and electrophysiological recordings show higher neural density near cardinal orientations — consistent with the efficient coding prediction.

0.164.2 12c.2 The Geometry of Tuning Curves

Consider a one-dimensional stimulus $s \in [0, 2\pi)$ (orientation) encoded by N neurons with Gaussian tuning curves:

$$f_i(s) = A \exp\left(-\frac{(s - \phi_i)^2}{2\sigma^2}\right),$$

where ϕ_i are the preferred orientations, uniformly spaced: $\phi_i = 2\pi i/N$. The population Fisher information at s is:

$$\mathcal{I}_{\text{pop}}(s) = \sum_{i=1}^{N} \frac{(f_i'(s))^2}{f_i(s)} = \frac{N}{2\pi} \int_0^{2\pi} \frac{(f'(s-\phi))^2}{f(s-\phi)}\, d\phi,$$

where the sum over neurons becomes an integral in the large-N limit, assuming uniform spacing $2\pi/N$. For Gaussian tuning, $\mathcal{I}_{\text{pop}}(s) = NA/(2\sigma^2)\cdot C$ (a constant independent of s, by the shift-invariant symmetry), so the Gaussian population code is an **isotropic** code: all stimuli are encoded with equal precision.

Correlated variability. When neurons exhibit correlated noise (positive noise correlations as typically observed), the population Fisher information is modified:

$$\mathcal{I}_{\text{pop}}^{\text{corr}}(s) = \mathbf{f}'(s)^\top \Sigma^{-1}(s) \mathbf{f}'(s),$$

where $\Sigma(s)$ is the noise covariance matrix. Correlations that are aligned with the signal dimension ("differential correlations" in the direction of $\mathbf{f}'(s)$) reduce Fisher information; correlations orthogonal to $\mathbf{f}'$ have no effect. This has the surprising implication that not all noise correlations are harmful: only those aligned with the readout direction degrade information transmission.

0.164.3 12c.3 Information-Limiting Correlations

Information-limiting correlations (Moreno-Bote et al., 2014) are a specific pattern of noise correlations that set a ceiling on the information extractable by any linear decoder, regardless of population size. Define $\mathbf{c}^{\text{info}} = \mathbf{f}'(s)/\|\mathbf{f}'(s)\|$ as the signal direction. Correlations of the form $\Sigma_{ij}^{\text{limit}} = \alpha c_i^{\text{info}} c_j^{\text{info}}$ (rank-1 correlations aligned with the signal) limit the Fisher information to at most $1/\alpha$ regardless of N:

$$\mathcal{I}_{\text{pop}}^{\text{limit}} = \frac{\|\mathbf{f}'(s)\|^2}{\|\mathbf{f}'(s)\|^2 \alpha + \text{diagonal terms}} \to \frac{1}{\alpha} \text{ as } N \to \infty.$$

This saturation of information with population size occurs in many cortical areas, suggesting that information-limiting correlations are a fundamental feature of neural coding. The cognitive implication: perceptual precision does not improve indefinitely with attention or learning; there is a ceiling determined by the information-limiting correlations in the relevant sensory area.

0.165 13d. Grid Cells, Entorhinal Cortex, and the Mathematics of Spatial Codes

0.165.1 13d.1 The Grid Cell Code

Entorhinal grid cells (Fyhn et al., 2004; Hafting et al., 2005) fire at multiple spatial locations arranged in a regular triangular lattice. Each grid cell is characterised by a **grid module**: a set of cells sharing the same lattice spacing (scale λ), orientation θ, and lattice phase $\phi \in [0, \lambda)^2$. A single grid module with N cells and random phases ϕ_i encodes position $\mathbf{r} \in \mathbb{R}^2$ as a population vector:

$$\mathbf{a}_\lambda(\mathbf{r}) = (f(\mathbf{r}; \phi_1), \ldots, f(\mathbf{r}; \phi_N)),$$

where $f(\mathbf{r}; \phi_i) = \sum_{j=1}^{3} \cos(\mathbf{k}_j \cdot (\mathbf{r} - \phi_i))$ is the firing rate function ($\mathbf{k}_j$ are the three wave vectors of the triangular lattice, $120°$ apart, with magnitude $4\pi/(\lambda\sqrt{3})$).

The **Fourier representation**: the grid code is a complex exponential code in two-dimensional spatial frequency space. The firing rate function $f(\mathbf{r}; \phi)$ has Fourier components only at the six lattice wave vectors $\pm\mathbf{k}_j$. The population vector $\mathbf{a}_\lambda(\mathbf{r})$ projects position onto a 6-dimensional subspace of the N-dimensional code space.

0.165.2 13d.2 Capacity and Decoding of Grid Codes

The number of distinct positions distinguishable by a single grid module with spacing λ is proportional to the grid period area λ^2 (the number of lattice cells in the environment). A **hierarchy** of K modules with spacings $\lambda_1 < \lambda_2 < \cdots < \lambda_K$ provides much larger capacity. If the spacings are in the ratio $\lambda_{k+1}/\lambda_k = r > 1$ (a geometric progression), the population code from all K modules can represent up to $\prod_{k=1}^{K} (\lambda_k/\sigma)^2$ distinguishable positions, where σ is the noise level.

For integer-ratio spacings r (so that $\lambda_K/\lambda_1 = r^{K-1}$ is a known integer), the grid hierarchy implements a **Chinese Remainder Theorem** (CRT) code: the phase of the grid at each scale provides one "digit" in a mixed-radix representation of position. The CRT guarantees that positions are uniquely represented by the phase tuple $(\phi_1, \ldots, \phi_K)$ as long as the position is within the range $[0, \mathrm{lcm}(\lambda_1, \ldots, \lambda_K))^2$. Decoding is equivalent to solving a system of congruences.

The information-theoretic capacity of the grid hierarchy: with K modules, each contributing $2\log_2(\lambda_k/\sigma)$ bits, the total capacity is:

$$C_{\text{grid}} = 2\sum_{k=1}^{K} \log_2(\lambda_k/\sigma) = 2\log_2\left(\prod_{k=1}^{K} \lambda_k/\sigma^K\right).$$

For a geometric hierarchy with $\lambda_k = \lambda_1 r^{k-1}$ and K modules, $C_{\text{grid}} = 2K\log_2(\lambda_1/\sigma) + K(K-1)\log_2 r$, growing quadratically in the number of modules.

0.165.3 13d.3 Path Integration and Attractor Dynamics

The entorhinal grid code is updated by **path integration**: as the animal moves with velocity $\mathbf{v}$, the grid cell phases shift continuously. The continuous attractor model (McNaughton et al., 2006) implements this as a neural dynamics:

$$r_i(\mathbf{r}) = f(\mathbf{r} - \phi_i), \quad \frac{d\phi_{\text{bump}}}{dt} = \mathbf{v}(t).$$

The "bump" of activity in the grid module shifts as a coherent wave, maintaining the current position estimate. The attractor is maintained by the periodic connectivity of the module: neurons with similar grid phases are recurrently connected, sustaining the pattern against noise.

The **error accumulation** in path integration is critical for cognitive function. Over time, noise accumulates in the phase estimate, leading to drift. After displacement d from the last landmark, the standard deviation of position error grows as $\sigma_{\text{drift}} \approx D\sqrt{d}$ (a random walk with diffusion constant D determined by the noise level and attractor strength). Landmark-based correction (visual inputs updating the phase estimate) resets the accumulating error, enabling precise long-range navigation.

0.166 14k. Reinforcement Learning: Mathematical Foundations

0.166.1 14k.1 Markov Decision Processes

A **Markov Decision Process** (MDP; Puterman 1994) is a tuple $(\mathcal{S}, \mathcal{A}, P, R, \gamma)$ where $\mathcal{S}$ is the state space, $\mathcal{A}$ the action space, $P(s'|s,a)$ the transition kernel, $R(s,a)$ the reward function, and $\gamma \in (0,1)$ the discount factor. A **policy** $\pi : \mathcal{S} \to \mathcal{P}(\mathcal{A})$ specifies the probability of each action in each state. The **value function** under policy π:

$$V^\pi(s) = \mathbb{E}^\pi\left[\sum_{t=0}^{\infty} \gamma^t R(S_t, A_t) \,\middle|\, S_0 = s\right],$$

satisfies the **Bellman equation**:

$$V^\pi(s) = \sum_a \pi(a|s)\left[R(s,a) + \gamma \sum_{s'} P(s'|s,a) V^\pi(s')\right].$$

The **optimal value function** $V^*(s) = \max_\pi V^\pi(s)$ satisfies the **Bellman optimality equation**:

$$V^*(s) = \max_a \left[R(s,a) + \gamma \sum_{s'} P(s'|s,a) V^*(s')\right],$$

a fixed-point equation that is the basis of dynamic programming. The **optimal policy** is greedy with respect to V^*: $\pi^*(s) = \arg\max_a [R(s,a) + \gamma \sum_{s'} P(s'|s,a) V^*(s')]$.

0.166.2 14k.2 Temporal Difference Learning

Temporal Difference (TD) learning (Sutton 1988) estimates the value function online from experience, without a model of P or R. The TD(0) update after observing transition (S_t, A_t, R_t, S_{t+1}):

$$\hat{V}(S_t) \leftarrow \hat{V}(S_t) + \alpha \underbrace{\left[R_t + \gamma \hat{V}(S_{t+1}) - \hat{V}(S_t)\right]}_{\delta_t \text{ (TD error)}},$$

where $\alpha \in (0,1)$ is the learning rate and δ_t is the **TD error** (the difference between the current value estimate and the bootstrapped target $R_t + \gamma \hat{V}(S_{t+1})$).

Theorem 0.166.1 (TD convergence; Dayan and Sejnowski 1994)**.** For a fixed policy π and step-size sequence satisfying $\sum_t \alpha_t = \infty$, $\sum_t \alpha_t^2 < \infty$, TD$(0)$ converges almost surely to V^π (with function approximation, to the TD fixed point in the linear function approximation case).

Q-learning (Watkins and Dayan 1992) is an off-policy TD method for learning the optimal action-value function $Q^*(s,a) = R(s,a) + \gamma \sum_{s'} P(s'|s,a) V^*(s')$:

$$\hat{Q}(S_t, A_t) \leftarrow \hat{Q}(S_t, A_t) + \alpha\left[R_t + \gamma \max_{a'} \hat{Q}(S_{t+1}, a') - \hat{Q}(S_t, A_t)\right].$$

Q-learning converges to Q^* regardless of the exploration policy, under mild conditions (Robbins-Monro step sizes, all state-action pairs visited infinitely often).

0.166.3 14k.3 The Dopamine-TD Correspondence

The neuroscientific significance of TD learning is the **dopamine-TD error hypothesis** (Schultz, Dayan, Montague 1997). Dopamine neurons in the ventral tegmental area (VTA) and substantia nigra fire in a pattern that matches the TD prediction error:

- **Unpredicted reward**: TD error $\delta_t = R_t > 0 \rightarrow$ dopamine burst.
- **Predicted reward received**: TD error $\delta_t = R_t + \gamma V(S_{t+1}) - V(S_t) \approx 0 \rightarrow$ no change.
- **Predicted reward omitted**: TD error $\delta_t = 0 + \gamma V(S_{t+1}) - V(S_t) < 0 \rightarrow$ dopamine dip.
- **Reward-predicting cue**: TD error $\delta_t = \gamma V(\text{cue state}) - V(\text{pre-cue state}) > 0 \rightarrow$ dopamine burst shifts to cue onset.

This correspondence is one of the most successful examples of a mathematical model explaining neural data quantitatively. It suggests that dopamine implements the teaching signal for value learning in the cortico-striatal circuit, with the striatum as the "critic" maintaining value estimates and the dopaminergic signal as the TD error that updates them.

0.166.4 14k.4 Deep Reinforcement Learning

Deep Q-Network (DQN; Mnih et al., 2015) replaces the tabular Q-function with a deep neural network $Q(s, a; \theta)$ and uses two key stabilisation tricks:

Experience replay: store transitions (S_t, A_t, R_t, S_{t+1}) in a replay buffer $\mathcal{D}$; update by sampling mini-batches uniformly from $\mathcal{D}$. This breaks temporal correlations and improves data efficiency.

Target network: maintain a separate "target" network $Q(s, a; \theta^-)$ with parameters θ^- updated less frequently than the main network θ. The TD target is computed using θ^-:

$$y_t = R_t + \gamma \max_{a'} Q(S_{t+1}, a'; \theta^-),$$

stabilising the training dynamics (without the target network, the moving target problem causes instability).

The DQN loss is:

$$\mathcal{L}(\theta) = \mathbb{E}_{(S,A,R,S') \sim \mathcal{D}} \left[\left(y - Q(S, A; \theta) \right)^2 \right],$$

minimised by stochastic gradient descent. DQN achieves superhuman performance on many Atari games using only raw pixel inputs, demonstrating the power of combining function approximation with RL.

Policy gradient methods. Instead of learning a value function and deriving a policy, policy gradient methods directly optimise the policy $\pi_\psi(a|s)$ parametrised by ψ. The **Policy Gradient Theorem** (Sutton et al., 1999):

$$\nabla_\psi J(\psi) = \mathbb{E}_{\pi_\psi} [\nabla_\psi \log \pi_\psi(A|S) \cdot Q^{\pi_\psi}(S, A)] \,,$$

where $J(\psi) = \mathbb{E}[V^{\pi_\psi}(S_0)]$ is the expected return. REINFORCE (Williams 1992) estimates this gradient using Monte Carlo returns: $\hat{G}_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k}$. Actor-critic methods combine a policy (actor) and value function (critic), using the critic to reduce the variance of the policy gradient estimator.

0.167 17e. Stochastic Processes in Cognition: A Synthesis

0.167.1 17e.1 The Ornstein-Uhlenbeck Process and Evidence Accumulation

The **Ornstein-Uhlenbeck** (OU) process is the continuous-time analogue of the leaky integrator, fundamental to perceptual decision-making:

$$dX_t = -\beta X_t\, dt + \sigma\, dW_t + \mu\, dt,$$

where μ is the drift (evidence rate), β is the leak (decay), σ is the noise amplitude, and W_t is standard Brownian motion. The OU process is the stationary Gaussian Markov process: its stationary distribution is $\mathcal{N}(\mu/\beta, \sigma^2/(2\beta))$.

Decision rule. A response is made when X_t first reaches a threshold $\pm a$ (the upper threshold for one choice, lower for the other). The joint distribution of response time T and choice C is characterised by the **first passage time** distribution of the OU process. For the special case $\beta = 0$ (no leak), this reduces to the classic diffusion model of Ratcliff (1978), for which the first passage time distribution has an inverse Gaussian form.

The leak-integration trade-off. The optimal value of β depends on the temporal structure of the evidence. For stationary evidence (constant drift throughout the trial), $\beta = 0$ is optimal: leaking evidence is suboptimal. For evidence that changes over time (non-stationary), a positive leak $\beta > 0$ downweights older (potentially stale) evidence, providing an optimal filter. The Bayesian optimal decision rule for a changing environment with transition rate λ requires $\beta = \lambda$ (the leak matches the change rate), showing that the integrator's time constant encodes the prior belief about temporal volatility.

0.167.2 17e.2 Diffusion Models: Analytical Results

For the pure diffusion model (Brownian motion with drift μ and noise σ, boundaries at $\pm a$ starting from $x_0 = 0$):

Probability of upper boundary hit:

$$P(\text{upper}) = \frac{\exp(2\mu a/\sigma^2) - 1}{\exp(2\mu a/\sigma^2) - \exp(-2\mu a/\sigma^2)} = \frac{1}{1 + \exp(-2\mu a/\sigma^2)}.$$

Mean decision time given upper boundary hit:

$$\mathbb{E}[T|\text{upper}] = \frac{a}{\mu}\tanh\!\left(\frac{\mu a}{\sigma^2}\right).$$

The RT distribution. The density of the first passage time to the upper boundary a is:

$$f_+(t) = \frac{a}{\sigma\sqrt{2\pi t^3}}\exp\!\left(-\frac{(a-\mu t)^2}{2\sigma^2 t}\right),$$

an inverse Gaussian density. This right-skewed distribution matches the empirical RT distributions of human and animal decision-making remarkably well.

Speed-accuracy trade-off. Reducing the boundary a decreases both mean RT and accuracy. The **optimal**

boundary maximises reward rate $R = P(\text{correct}) \cdot r / (E[T] + T_{\text{ITI}})$ where r is reward per correct response and T_{ITI} is inter-trial interval. Differentiating with respect to a gives the optimal boundary equation:

$$\frac{\partial \log R}{\partial a} = 0 \Rightarrow \frac{\partial \log P(\text{correct})/\partial a}{P(\text{correct})} = \frac{\partial E[T]/\partial a}{E[T] + T_{\text{ITI}}}.$$

For simple detection tasks, the optimal boundary is roughly $a^* \approx \sigma\sqrt{T_{\text{ITI}}\mu}$, increasing with inter-trial interval and decreasing with evidence quality μ/σ.

0.167.3 17e.3 Point Processes and Spike Train Statistics

Point processes model sequences of discrete events (spikes) in continuous time. The **Poisson process** with rate $\lambda > 0$ generates events in $[t, t+dt)$ with probability $\lambda\, dt + O(dt^2)$, independently across disjoint intervals. Spike counts in $[0, T]$ follow Poisson(λT); inter-spike intervals follow Exp(λ).

The **log-Gaussian Cox process** is a more realistic model: the instantaneous rate $\lambda(t)$ is itself a random process (modulated by sensory input, attention, adaptation). If $\log \lambda(t) \sim \mathcal{GP}(m, k)$, the resulting spike train is a doubly stochastic Poisson process with extra-Poisson variability (Fano factor $F = \text{Var(count)}/\text{Mean(count)} > 1$, as observed in cortical neurons).

Fano factor and neural variability. The Fano factor for a Poisson process is exactly 1. For cortical neurons, $F \approx 1$–3 for most areas, but reaches $F \approx 0.5$ for neurons with strong stimulus drive (quenching variability through the nonlinear gain function). The **variance-to-mean ratio** provides a sensitive test for deviations from Poisson statistics: overdispersion ($F > 1$) suggests additional noise sources (synaptic fluctuations, network dynamics); underdispersion ($F < 1$) indicates inhibitory control or subthreshold integration.

0.168 18i. Causal Discovery Algorithms: Complete Treatment

0.168.1 18i.1 The PC Algorithm

The **PC algorithm** (Peters, Janzing, Schölkopf 2017) recovers the **Markov equivalence class** (MEC) of the data-generating DAG $\mathcal{G}$ from observational data, under the faithfulness assumption (all conditional independences in the data are entailed by $\mathcal{G}$).

Phase 1: Skeleton identification. Start with a complete undirected graph. For each pair of variables (X_i, X_j), test conditional independence $X_i \perp\!\!\!\perp X_j | S$ for subsets S of increasing size. If $X_i \perp\!\!\!\perp X_j | S$ for some S, remove the edge $X_i - X_j$ and record $\text{Sep}(X_i, X_j) = S$.

Phase 2: V-structure orientation. For each unshielded triple $X_i - X_k - X_j$ (with no edge $X_i - X_j$), orient as $X_i \to X_k \leftarrow X_j$ (**v-structure**) iff $X_k \notin \text{Sep}(X_i, X_j)$.

Phase 3: Orientation propagation. Apply the Meek rules to orient as many remaining edges as possible without creating new v-structures or directed cycles.

Statistical implementation. Conditional independence tests are typically based on partial correlations (for Gaussian data, $X_i \perp\!\!\!\perp X_j | S$ iff the partial correlation $\rho_{ij|S} = 0$, testable via Fisher's z-transform) or kernel-based tests (HSIC, KCI) for non-linear dependencies.

Sample complexity. The PC algorithm's sample complexity for consistent recovery of the MEC scales as $O(n^{2q} \log p)$ where n is the sample size, p is the number of variables, and q is the maximum degree (number of adjacent nodes) in the skeleton. For sparse graphs with bounded degree, this gives polynomial sample complexity in p.

0.168.2 18i.2 Score-Based Methods: GES and FGES

Greedy Equivalence Search (GES; Chickering 2002) operates directly on the MEC space, searching for the MEC with highest score. GES has three phases:

Phase 1 (Forward search): Start from the empty graph. Greedily add edges (as completed PDAGs) that maximally increase the score until no edge can increase it.

Phase 2 (Backward search): Greedily remove edges that maximally increase the score until no edge can increase it.

Phase 3 (Turning): Turn undirected edges to orient remaining edges without changing the score.

For **decomposable scores** (BIC, BDeu), each score function decomposes as $S(\mathcal{G}) = \sum_{i=1}^{p} S(X_i, \text{PA}_i)$: the score is a sum of local scores, one per variable. This allows efficient computation of score changes when edges are added or removed. GES is consistent for Gaussian linear models (Chickering 2002) and for general faithful distributions with a compatible decomposable score.

FGES (Ramsey et al., 2017) speeds up GES by reusing computed scores in a priority queue and distributing the score computation, making it applicable to datasets with thousands of variables.

0.168.3 18i.3 Functional Causal Models and Non-Gaussianity

For linear Gaussian models, the MEC is the finest identifiable structure (the exact DAG cannot be recovered from observational data alone). However, for **non-Gaussian** or **nonlinear** models, the causal direction is identifiable:

LiNGAM (Linear Non-Gaussian Acyclic Model; Shimizu et al., 2006): if the data are generated by a linear DAG with non-Gaussian noise, the causal ordering is identifiable. The algorithm uses Independent Component Analysis (ICA) to identify the mixing matrix B (where $\mathbf{x} = B\mathbf{x} + \boldsymbol{\varepsilon}$) and the causal ordering is determined by the matrix $I - B$.

ANM (Additive Noise Model; Hoyer et al., 2009): for a bivariate model $Y = f(X) + N$ with $N \perp X$, the causal direction $X \to Y$ is identifiable if f is nonlinear (or N is non-Gaussian). The reverse model $X = g(Y) + M$ with $M \perp Y$ is generically not achievable: testing independence of residuals from both models identifies the causal direction.

Cognitive causal discovery. These methods have been applied to identify causal relationships in cognitive and neuroscientific data: does working memory capacity causally determine reasoning performance (rather than vice versa)? Does early sensory processing causally affect later decision-related activity? The non-Gaussianity of neural signals (and of many cognitive variables) makes LiNGAM and ANM particularly suitable for cognitive data.

0.169 19g. Statistical Learning Theory: Generalisation Bounds

0.169.1 19g.1 VC Dimension and Uniform Convergence

Statistical learning theory (Vapnik 1998; Shalev-Shwartz and Ben-David 2014) provides upper bounds on the generalisation error of learning algorithms. The key quantity is the **Vapnik-Chervonenkis (VC) dimension** of a hypothesis class $\mathcal{H}$.

Definition 0.169.1 (VC dimension)**.** A set $C = \{x_1, \ldots, x_m\}$ is **shattered** by $\mathcal{H}$ if for every labelling $\mathbf{y} \in \{0,1\}^m$, there exists $h \in \mathcal{H}$ with $h(x_i) = y_i$ for all i. The VC dimension $\text{VC}(\mathcal{H}) = \max\{m : \exists C \subseteq \mathcal{X},\, |C| = m,\, C \text{ shattered by } \mathcal{H}\}$.

Theorem 0.169.2 (Fundamental theorem of statistical learning; Blumer et al. 1989)**.** A hypothesis class $\mathcal{H}$ is PAC-learnable iff $\text{VC}(\mathcal{H}) < \infty$. Furthermore, for any $\epsilon, \delta > 0$, after $n \geq C \cdot (\text{VC}(\mathcal{H}) + \log(1/\delta))/\epsilon$ examples, any ERM learner achieves error $\leq \epsilon$ with probability $\geq 1 - \delta$.

VC dimensions of common hypothesis classes:

- Linear classifiers in $\mathbb{R}^d$: $\text{VC} = d + 1$.
- Axis-aligned rectangles in $\mathbb{R}^d$: $\text{VC} = 2d$.
- Neural networks (depth L, width W, ReLU activations): $\text{VC} = O(LW^2 \log W)$ (Bartlett et al., 2019).
- Gaussian kernel SVM: $\text{VC} = \infty$ (infinite-dimensional RKHS), but Rademacher complexity provides tighter bounds.

Rademacher complexity. A tighter data-dependent measure of complexity: the **empirical Rademacher complexity** of $\mathcal{H}$ on sample $S = (x_1, \ldots, x_n)$ is:

$$\hat{\mathfrak{R}}_S(\mathcal{H}) = \mathbb{E}_\sigma \left[\sup_{h \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \sigma_i h(x_i) \right],$$

where $\sigma_i \in \{-1, +1\}$ are i.i.d. Rademacher variables. The generalisation bound: with probability $\geq 1 - \delta$, for all $h \in \mathcal{H}$:

$$L(h) \leq \hat{L}_S(h) + 2\hat{\mathfrak{R}}_S(\mathcal{H}) + \sqrt{\frac{\log(1/\delta)}{2n}},$$

where $L(h)$ is the true risk and $\hat{L}_S(h)$ is the empirical risk. Rademacher complexity is tighter than VC-based bounds because it adapts to the specific sample distribution.

0.169.2 19g.2 Margin Bounds and Deep Networks

For classification with margin $\gamma > 0$, the sample complexity depends on the margin: larger margin implies better generalisation. The **margin bound** for linear classifiers in RKHS:

$$L(h) \leq \hat{L}_S^\gamma(h) + O\left(\frac{\|h\|_{\mathcal{H}}}{\gamma \sqrt{n}} \right),$$

where $\hat{L}_S^\gamma$ is the fraction of training examples with margin $< \gamma$. For SVMs, the norm $\|h\|_{\mathcal{H}}$ equals $\|w\|$ (the weight vector norm), and the bound depends on the maximum kernel value $\sup_x k(x,x)$, not on the

ambient dimension.

Double descent. Modern deep networks often have $\text{VC}(\mathcal{H}) \gg n$ (many more parameters than data points), yet generalise well. The **double descent** phenomenon (Belkin et al., 2019; Hastie et al., 2019): as model complexity increases, the test error first decreases (classical bias-variance trade-off), then increases (overfitting), then decreases again (interpolation regime) as the model becomes large enough to both interpolate training data and generalise. The implicit regularisation of gradient descent (in the minimum-norm interpolator for linear models, and approximately for neural networks) explains the second descent.

Cognitive implications. The learnability theory connects to the Bayesian word learning model (Section 3k): PAC-learnability with sample complexity $O(1/\epsilon)$ for VC dimension 1 (nested hypothesis classes) explains why children can learn new words from very few examples. The VC dimension of the child's concept space determines the sample complexity, providing a quantitative link between the richness of the representational system and the number of examples needed for acquisition.

0.170 20q. Cognitive Complexity and Algorithmic Information

0.170.1 20q.1 Kolmogorov Complexity and Cognitive Simplicity

The **Kolmogorov complexity** $K(x)$ of a finite string x is the length of the shortest program that outputs x on a universal Turing machine U:

$$K(x) = \min_{p: U(p)=x} |p|.$$

K is **universal**: for any other universal Turing machine U', $K_{U'}(x) \leq K_U(x) + C_{U,U'}$ (the constant C depends only on U and U', not on x). Kolmogorov complexity is therefore machine-independent up to a constant.

The randomness deficiency $d(x) = \log |X| - K(x)$ measures how much simpler x is than a random string of the same length. Objects with $d(x) \approx 0$ are algorithmically random; those with large $d(x)$ are simple (compressible).

Cognitive simplicity. The algorithmic complexity framework models human perception of simplicity: patterns perceived as "simple" by humans correspond to strings with low Kolmogorov complexity. The **Simplicity Theory** (Chater and Vitányi 2003; Feldman 2016) proposes that cognitive parsing selects the interpretation with minimum description length (MDL):

$$\hat{I} = \arg\min_{I \in \mathcal{I}} K(I) + K(D|I),$$

where $\mathcal{I}$ is the set of candidate interpretations and $K(D|I)$ is the conditional complexity of the data given interpretation I. This is formally equivalent to MAP inference with a prior $P(I) \propto 2^{-K(I)}$ (the **universal prior** of Solomonoff-Levin): interpretations are preferred in inverse proportion to their description length.

0.170.2 20q.2 Descriptive Complexity and Cognitive Grammars

Descriptive complexity (Immerman 1999; Fagin 1974) characterises the expressive power of logical fragments by the computational complexity class they define. Key results:

- **FO (first-order logic)**: defines exactly the properties in AC^0 (constant-depth circuits) over ordered

structures.

- **SO (second-order logic)**: defines exactly NP (Fagin's theorem, 1974).
- **FO+LFP (least fixed point)**: defines exactly PTIME over ordered structures (Immerman-Vardi theorem, 1987).
- **FO+PFP (partial fixed point)**: defines exactly PSPACE.

For cognitive science, these results suggest that the complexity of cognitive tasks is constrained by the logical expressiveness of the language of thought. If the language of thought has the expressiveness of FO+LFP, cognitive computations are PTIME; if it is full second-order, they may be NP-complete. The **tractability hypothesis** (van Rooij 2008) proposes that cognitive computations are actually computable in polynomial time, suggesting that cognitive algorithms exploit special structure (e.g., fixed-parameter tractability) to avoid worst-case exponential complexity.

0.170.3 20q.3 Minimum Description Length in Practice

The **Minimum Description Length** (MDL) principle (Rissanen 1978, 1996) selects the model that minimises the two-part description:

$$\text{MDL}(M, D) = L(M) + L(D|M),$$

where $L(M)$ is the code length of the model and $L(D|M)$ is the code length of the data given the model. For a model class $\mathcal{M}$ with parameter θ:

$$\text{MDL}(\theta) \approx -\log p(D|\hat{\theta}) + \frac{k}{2}\log n,$$

where k is the number of parameters and n is sample size. This is the BIC (Section 3d.2), which is the large-sample approximation to MDL. The exact MDL uses the **Normalised Maximum Likelihood** (NML):

$$L_{NML}(D, \mathcal{M}) = -\log \frac{p(D|\hat{\theta}(D))}{\sum_{D'} p(D'|\hat{\theta}(D'))},$$

where the denominator sums over all possible datasets, making NML the minimax optimal two-part code.

MDL for cognitive grammar learning. Solan et al. (2005) applied MDL to language acquisition, proposing the ADIOS algorithm: extract a grammar from a corpus by iteratively finding the compression that minimises description length, merging equivalent contexts and abstracting slot-filling patterns. The resulting grammar captures recursive structure from flat text without explicit supervision, providing a formal model of unsupervised grammar induction from positive examples alone.

0.171 21g. Topological Data Analysis and Cognitive Representations

0.171.1 21g.1 Persistent Homology

Persistent homology (Edelsbrunner et al., 2002; Zomorodian and Carlsson 2005) computes the topological features (connected components, loops, voids) of a point cloud at multiple scales, tracking which features persist across scales.

Given a point cloud $X = \{x_1, \ldots, x_n\} \subset \mathbb{R}^d$, the **Vietoris-Rips complex** at scale ϵ includes a simplex $\{x_{i_0}, \ldots, x_{i_k}\}$ iff all pairwise distances are $\leq \epsilon$. As ϵ increases from 0 to ∞, simplices are added, and topological features (homology classes) are **born** and **die**. The **persistence diagram** records birth-death pairs (b_k, d_k) for each homological dimension; long-lived features (large $d_k - b_k$) are considered topologically significant.

Stability theorem. The persistence diagram is stable under small perturbations of the point cloud: $d_B(\text{PD}(X), \text{PD}(Y)) \leq d_{GH}(X, Y)$, where d_B is the bottleneck distance between diagrams and d_{GH} is the Gromov-Hausdorff distance between point clouds. This makes persistent homology a robust geometric descriptor for noisy data.

0.171.2 21g.2 Topological Analysis of Neural Representations

Neural manifold geometry can be analysed using persistent homology. If neural representations lie on a low-dimensional manifold embedded in high-dimensional firing rate space, the topology of that manifold encodes the structure of the represented space.

Place cell topology. Curto and Itskov (2008) showed that the firing patterns of hippocampal place cells encode the topology of the environment. A **clique complex** is built from co-active cell populations: a simplex $\{c_{i_0}, \ldots, c_{i_k}\}$ is included iff cells $c_{i_0}, \ldots, c_{i_k}$ all fire simultaneously. The homology of this complex reflects the topology of the environment: a circular environment produces a H_1 class (a 1-cycle); a torus (as in toroidal virtual environments) produces two H_1 classes and one H_2 class.

Representational topology of visual cortex. Carlsson et al. (2008) applied persistent homology to natural image patches (small patches from natural images). The topological analysis revealed that the space of natural image patches has the topology of a Klein bottle: not a sphere, not a torus, but a non-orientable surface. This implies that low-level visual representations exploit the non-orientable topology of natural image statistics, providing a mathematical constraint on the structure of simple-cell filters in V1.

0.171.3 21g.3 Mapper Algorithm for Brain Networks

The **Mapper algorithm** (Singh et al., 2007) constructs a topological summary of a high-dimensional dataset as a simplicial complex:

1. Choose a **filter function** $f : X \to \mathbb{R}^k$ (a dimensionality reduction or lens function).
2. Cover the range of f with overlapping intervals $U_1, \ldots, U_m$.
3. For each U_j, cluster the pre-image $f^{-1}(U_j)$.
4. Build a simplicial complex: vertices are clusters; edges connect clusters that share data points; higher simplices for larger overlaps.

The Mapper graph reveals the **global shape** of the data: branching points, loops, flares. Applied to fMRI time series with a functional connectivity filter, Mapper reveals the topological structure of brain state transitions: how cognitive states (rest, task, attention) are connected in the space of brain activity patterns, and which topological features (loops = cyclic state transitions, branches = state bifurcations) characterise each condition.

0.172 22k. Probabilistic Semantics and Meaning Uncertainty

0.172.1 22k.1 Probabilistic Type Theory

Probabilistic Type Theory (ProbTT; Goodman et al., 2015; Cooper et al., 2015) extends the typed lambda calculus (Section 22h) with probability distributions over semantic denotations. Instead of a word w having a fixed denotation $[\![w]\!] \in \mathcal{D}$, it has a **probabilistic denotation**: a distribution over possible meanings $P_w : \mathcal{D} \to [0, 1]$.

For example, the adjective "tall" does not have a fixed threshold; instead, it expresses a context-dependent predicate whose threshold is uncertain: $P(\text{tall}(x)) = P(h(x) > \theta | \theta \sim \text{context})$, where θ is drawn from a prior calibrated to the context (population of reference). This is the **Bayesian analysis of vagueness** (Lassiter and Goodman 2013): the meaning of gradable adjectives involves uncertainty over the comparison standard.

Semantic composition under uncertainty. Given probabilistic denotations for words, the denotation of a phrase is the **lifted composition**: if $[\![f]\!] \sim P_f$ and $[\![a]\!] \sim P_a$, then $[\![f(a)]\!]$ is the distribution over P_f-sampled functions applied to P_a-sampled arguments:

$$P_{f(a)}(d) = \int\int P_f(f)\, P_a(a)\, \delta(f(a) = d)\, df\, da.$$

This is functorial: it corresponds to applying the applicative functor structure of probability distributions to the semantic composition operations of the grammar.

0.172.2 22k.2 Lexical Semantics: Distributional Models

Distributional semantics (Harris 1954; Turney and Pantel 2010) represents word meaning as a vector in a high-dimensional space derived from co-occurrence statistics. The **distributional hypothesis**: words that occur in similar contexts have similar meanings. Formally, define a word-context matrix $M \in \mathbb{R}^{|V| \times |C|}$ where M_{wc} is the frequency of word w in context c (e.g., within a window of ± 5 words). The word vector $\mathbf{v}_w = M_w$ is the word's semantic representation.

PMI and PPMI. Raw co-occurrence counts are dominated by frequent words; the **Pointwise Mutual Information** weighting corrects for this:

$$\text{PMI}(w, c) = \log \frac{P(w, c)}{P(w)P(c)} = \log \frac{M_{wc} \cdot N}{\sum_c M_{wc} \cdot \sum_w M_{wc}},$$

where $N = \sum_{w,c} M_{wc}$. **Positive PMI** (PPMI) truncates negative values at 0. Word2Vec (Mikolov et al., 2013) and GloVe (Pennington et al., 2014) are neural implementations that implicitly approximate PPMI-weighted matrix factorisation, producing dense vector representations that capture semantic regularities (e.g., $\mathbf{v}_{\text{king}} - \mathbf{v}_{\text{man}} + \mathbf{v}_{\text{woman}} \approx \mathbf{v}_{\text{queen}}$).

Geometric semantics. The vector space of word embeddings has geometric structure that corresponds to semantic structure:

- **Cosine similarity** $\cos(\mathbf{v}_w, \mathbf{v}_{w'}) = \mathbf{v}_w \cdot \mathbf{v}_{w'} / (\|\mathbf{v}_w\| \|\mathbf{v}_{w'}\|)$ correlates with semantic similarity (as judged by humans).

- **Analogical proportions** ($a : b :: c :?$) correspond to vector arithmetic: $\hat{\mathbf{v}}_? \approx \mathbf{v}_b - \mathbf{v}_a + \mathbf{v}_c$ (the 3CosAdd method).
- **Semantic categories** correspond to clusters in the embedding space, identifiable by k-means or spectral clustering.

0.172.3 22k.3 Transformer Architecture and Attention

The **transformer architecture** (Vaswani et al., 2017) is the computational backbone of large language models. Its key component is **scaled dot-product attention**:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V,$$

where $Q = XW_Q$, $K = XW_K$, $V = XW_V$ are the query, key, and value matrices (linear projections of the input X), and d_k is the key dimension. The softmax normalises the attention weights so that each output token is a convex combination of value vectors, weighted by the similarity (dot product) between the query and each key.

Multi-head attention runs H parallel attention operations ("heads") with different projection matrices, concatenating the outputs:

$$\text{MHA}(Q, K, V) = \text{concat}(\text{head}_1, \ldots, \text{head}_H)W_O,$$

$$\text{head}_h = \text{Attention}(QW_{Q,h}, KW_{K,h}, VW_{V,h}).$$

Different heads specialise to different types of attention (syntactic, semantic, long-range, local), providing a distributed representation of contextual relationships.

Information-theoretic analysis. The attention mechanism implements a form of **content-addressable memory**: the query q_i retrieves the most relevant values from the key-value pairs. The information-theoretic capacity of an attention layer (Bhojanapalli et al., 2020) is bounded by the entropy of the attention distribution: $H(\text{attn}) = -\sum_j \alpha_j \log \alpha_j$. High-entropy attention (uniform) retrieves an average; low-entropy attention (peaked) retrieves a specific value. This parallels the precision-weighted inference of predictive coding: high-precision retrieval (low entropy attention) corresponds to confident context-dependent processing.

0.173 23e. Embodied Mathematics: Enactive Cognition and Dynamical Systems

0.173.1 23e.1 Enactivism and Sensorimotor Contingencies

Enactivism (Varela, Thompson, Rosch 1991; O'Regan and Noë 2001) proposes that cognition is constituted by sensorimotor activity, not by internal representations. The brain does not construct an inner model of the world; it participates in ongoing sensorimotor loops that couple the agent to its environment.

The **sensorimotor contingency** (SMC) framework formalises this: for each sensory modality, there is a set of **sensorimotor laws** ϕ : Motor commands $\times$ World states $\to$ Sensory signals governing how motor actions change sensory input. Perception is the mastery of these laws: to see is to have implicit

knowledge of the visual SMCs (what visual changes will result from each eye movement, head rotation, or body movement).

Mathematical formalisation. Let S be the sensory state space, M the motor state space, and W the world state space. Define the SMC map $\phi : M \times W \to S$. An agent that acts with a specific motor command m in world state w produces sensory change $\Delta s = \phi(m, w) - s$. The **sensorimotor manifold** is the image $\phi(M, W) \subset S$: the set of sensory states accessible through all motor actions in all world states.

The equivalence class of world states that are indistinguishable via SMCs defines the agent's **perceptual categories**: states w and w' are perceptually equivalent iff $\phi(M, w) = \phi(M, w')$ as functions of M. The quotient space $W/\sim_\phi$ is the agent's perceptual space, which is a function of both world structure and motor apparatus, not of world structure alone.

0.173.2 23e.2 Dynamical Systems Theory of Coordination

Dynamical systems theory (Kelso 1995; Turvey 1990) models cognitive and motor behaviour as trajectories in a continuous state space governed by differential equations. The emphasis is on **self-organisation**: complex coordinated behaviour emerges from the interaction of simpler subsystems without explicit central control.

Haken-Kelso-Bunz model. The HKB model (Haken et al., 1985) describes the dynamics of bimanual coordination in terms of the relative phase ϕ between the two hands. The model is a gradient system:

$$\dot{\phi} = -\nabla V(\phi), \quad V(\phi) = -a\cos\phi - b\cos 2\phi,$$

where $a > 0$ and b depends on the movement frequency. The potential V has two minima: $\phi = 0$ (in-phase, both hands move in the same direction) and $\phi = \pi$ (anti-phase, hands move in opposite directions). As movement frequency increases, b/a decreases; above a critical frequency the anti-phase minimum disappears (a saddle-node bifurcation), and the system transitions to in-phase coordination. This **nonequilibrium phase transition** is the formal signature of a self-organised switching phenomenon, explained entirely by the landscape of V without invoking a decision mechanism.

Attractors and cognitive state transitions. More generally, cognitive states can be modelled as attractors of a dynamical system:

$$\dot{\mathbf{x}} = F(\mathbf{x}) + \boldsymbol{\xi}(t),$$

where $\mathbf{x}$ is the cognitive state vector, F is the deterministic dynamics, and $\boldsymbol{\xi}$ is noise. A cognitive state is an attractor $\mathbf{x}^*$ of F: a stable fixed point ($\mathrm{Re}(\lambda_i(\nabla F(\mathbf{x}^*))) < 0$ for all eigenvalues λ_i), a limit cycle, or a strange attractor (for chaotic dynamics). Transitions between cognitive states correspond to **noise-induced transitions** between attractors, whose rate is governed by Kramers' escape rate formula:

$$\tau_{\text{escape}} \propto \exp\left(\frac{\Delta V}{D}\right),$$

where ΔV is the potential barrier height and D is the noise amplitude. Precision (the inverse of noise) thus controls the stability of cognitive states: high precision (low noise) leads to long dwell times in attractor states (stability of mental sets); low precision (high noise) facilitates rapid transitions (cognitive flexibility).

0.173.3 23e.3 Thermodynamics of Cognition and Landauer's Principle

Landauer's principle (Landauer 1961; Bennett 1973) connects information processing to thermodynamics: the minimum heat dissipated when erasing one bit of information is $k_B T \ln 2$ (where k_B is Boltzmann's constant and T is temperature). This is the fundamental thermodynamic cost of irreversible computation.

Cognitive implications. If the brain implements reliable computation, neural information erasure (e.g., resetting a working memory representation) requires dissipation of at least $k_B T \ln 2 \approx 3 \times 10^{-21}$ J per bit at physiological temperature. The brain dissipates approximately 20 W, corresponding to about 5×10^{21} operations per second at the Landauer limit — far more than the estimated 10^{17} synaptic operations per second, suggesting a factor of 10^4 inefficiency relative to the thermodynamic limit. This gap motivates the study of neuromorphic computing: hardware architectures that approach the Landauer limit by implementing spike-based, event-driven computation.

Free energy and thermodynamic free energy. The variational free energy of the FEP (Section IV) is formally analogous to the Helmholtz free energy $F = U - TS$ of statistical mechanics. The analogy: $U \leftrightarrow -\mathbb{E}_q[\log p(o, s)]$ (internal energy = negative expected log-joint), $S \leftrightarrow H(q)$ (entropy of recognition distribution). Minimising variational free energy is analogous to minimising thermodynamic free energy in a non-equilibrium process: both drive the system toward the lowest free energy state consistent with the constraints, which for the variational case is the posterior $q^* = p(s|o)$.

0.174 24d. Category Theory and Cognition: A Deeper Look

0.174.1 24d.1 Adjunctions and Cognitive Processes

An **adjunction** $F \dashv G$ between categories $\mathcal{C}$ and $\mathcal{D}$ consists of functors $F : \mathcal{C} \to \mathcal{D}$ and $G : \mathcal{D} \to \mathcal{C}$ with a natural bijection $\mathcal{D}(F(A), B) \cong \mathcal{C}(A, G(B))$. Adjunctions are ubiquitous in mathematics and have been proposed as a structure capturing cognitive processes of **generalisation** (going from A to $G(B)$) and **specification** (going from $F(A)$ to B).

The Free-Forgetful Adjunction. The free functor F : **Set** $\to$ **Grp** (sending a set to the free group it generates) is left adjoint to the forgetful functor G : **Grp** $\to$ **Set** (forgetting the group structure). This adjunction captures the relationship between unstructured sets and structured groups: to give a set map $A \to G(H)$ (a function from the generators into the group H) is the same as to give a group homomorphism $F(A) \to H$ (extending the map to all group elements by the freeness property). In cognitive terms, acquiring structured knowledge (extending from examples to a grammar) mirrors the left adjoint, while applying knowledge in unstructured contexts mirrors the right adjoint.

Kan extensions as optimal approximations. The **left Kan extension** $\mathrm{Lan}_K F$ of a functor $F : \mathcal{A} \to \mathcal{C}$ along a functor $K : \mathcal{A} \to \mathcal{B}$ is the best approximation of F that factors through K:

$$(\mathrm{Lan}_K F)(B) = \mathrm{colim}_{K(A) \to B} F(A),$$

where the colimit is over the comma category $(K \downarrow B)$. Kan extensions generalise along an incomplete specification: if F is defined on a small category $\mathcal{A}$ (observed cases), $\mathrm{Lan}_K F$ extends it to a larger category $\mathcal{B}$ (all cases) by taking the "best left-approximation." This is the categorical analogue of Bayesian generalisation: Kan extension from observed cases to unobserved cases minimises (colimit

= left-approximation) the mismatch.

0.174.2 24d.2 Monads and Computational Effects

A **monad** (T, η, μ) on a category $\mathcal{C}$ consists of a functor $T : \mathcal{C} \to \mathcal{C}$, a natural transformation $\eta : \mathrm{Id} \Rightarrow T$ (unit), and a natural transformation $\mu : T^2 \Rightarrow T$ (multiplication), satisfying the monad laws:

$$\mu \circ T\mu = \mu \circ \mu T \quad \text{(associativity)}, \qquad \mu \circ T\eta = \mu \circ \eta T = \mathrm{id}_T \quad \text{(unit laws)}.$$

The probability monad. The **distribution monad** $\mathcal{D}$ on **Meas** (measurable spaces) sends a measurable space X to the space of probability distributions $\mathcal{D}(X)$. The unit $\eta_X : X \to \mathcal{D}(X)$ sends $x \mapsto \delta_x$ (Dirac delta). The multiplication $\mu_X : \mathcal{D}(\mathcal{D}(X)) \to \mathcal{D}(X)$ marginalises out the outer distribution (it is the expectation of the inner distribution). A **Kleisli morphism** $f : X \to \mathcal{D}(Y)$ is a random function (a stochastic map), and Kleisli composition $f \circ_{\mathcal{D}} g = \mu_Y \circ \mathcal{D}(f) \circ g$ is the probabilistic chain rule. The Kleisli category of the distribution monad is the category of Markov kernels, fundamental to probabilistic programming and Bayesian inference.

Cognitive interpretation. The monad structure provides a compositional account of cognitive processes with uncertainty. A cognitive transformation that takes an input observation and produces a probability distribution over possible interpretations is a Kleisli morphism. Composing two such transformations (e.g., a phonemic parser followed by a lexical parser) is Kleisli composition: it integrates the uncertainty from both stages. The monad laws ensure that the composition is well-defined and associative, providing a mathematical foundation for **compositional probabilistic cognition**.

0.174.3 24d.3 Sheaves and Contextuality

A **sheaf** on a topological space X assigns to each open set $U \subseteq X$ a set (or algebraic structure) $\mathcal{F}(U)$ (the "sections over U") and restriction maps $\rho_{UV} : \mathcal{F}(U) \to \mathcal{F}(V)$ for $V \subseteq U$, satisfying a compatibility (gluing) condition: sections that agree on overlaps can be uniquely glued into a global section.

Sheaves in cognitive science. Abramsky and Brandenburger (2011) used sheaf theory to unify quantum contextuality and classical probabilistic phenomena. A **contextuality scenario** is a set of measurements with overlapping contexts; a **no-signalling behaviour** is a sheaf of probability distributions over measurement outcomes that is consistent on overlaps but cannot be glued to a global distribution. The **Contextuality-by-Default** (CbD) framework (Dzhafarov et al., 2016) extends this to psychology: cognitive measurements made in different contexts are modelled as sections of a sheaf, and violations of sheaf-theoretic consistency (non-extendability) indicate genuine contextuality in human judgment.

The sheaf-theoretic obstruction to globalisation is measured by the Čech cohomology of the scenario: the first cohomology group H^1 contains the obstructions. A zero first cohomology group means local distributions glue to a global one (no contextuality); non-zero H^1 indicates contextuality. This provides a computable topological invariant for the degree of contextuality in a cognitive or quantum system.

0.175 25e. Mathematical Models of Social Cognition

0.175.1 25e.1 Theory of Mind and Higher-Order Beliefs

Theory of Mind (ToM) is the ability to attribute mental states (beliefs, desires, intentions) to others. Mathematically, ToM requires representing beliefs about beliefs: **higher-order beliefs**. A first-order belief is $B_A(P)$ (agent A believes P); a second-order belief is $B_A(B_B(P))$ (agent A believes that agent B believes P); and so on.

Recursive Bayesian ToM. The **Rational Speech Acts** (RSA) model (Frank and Goodman 2012; Goodman and Frank 2016) formalises pragmatic language understanding as a recursive Bayesian inference:

$$P_{L_0}(m|u) \propto [\![u]\!](m), \quad P_{S_k}(u|m) \propto \exp(\lambda \log P_{L_{k-1}}(m|u)), \quad P_{L_k}(m|u) \propto P_{S_k}(u|m)P(m).$$

The literal listener L_0 interprets utterance u as true. The pragmatic speaker S_k chooses u to maximise the posterior probability that the literal listener L_{k-1} infers the intended meaning m. The pragmatic listener L_k inverses the pragmatic speaker. Recursion converges to a fixed point for $k \to \infty$ (assuming λ is not too large), approximating the rational agent's interpretation.

Experimental validation. RSA predicts:

- **Scalar implicature**: "Some students passed" $\Rightarrow$ "Not all passed" (the implicature arises because the speaker would have said "all" if true; the S_1 speaker's choice of "some" signals that "all" is false).
- **Reference games**: given a context with three objects, "the blue circle" refers to the only blue circle even if there is another blue object, because the speaker would have been more specific if the intended referent were ambiguous.
- **Irony**: the pragmatic interpretation can flip the literal meaning when the prior favours the opposite interpretation.

0.175.2 25e.2 Game Theory and Strategic Cognition

Bayesian games (Harsanyi 1967, 1968) model strategic interaction under uncertainty about other agents' types. A Bayesian game is a tuple $(\mathcal{N}, \mathcal{A}, \mathcal{T}, P, U)$ where $\mathcal{N}$ is the set of players, $\mathcal{A}_i$ is player i's action space, $\mathcal{T}_i$ is player i's type space, $P(\mathbf{t})$ is the joint type distribution (common prior), and $U_i : \mathcal{A} \times \mathcal{T} \to \mathbb{R}$ is the utility function. A **Bayesian Nash Equilibrium** (BNE) is a strategy profile $\sigma_i : \mathcal{T}_i \to \mathcal{P}(\mathcal{A}_i)$ such that each player maximises expected utility given their type and beliefs:

$$\sigma_i^*(t_i) = \arg\max_{a_i} \mathbb{E}_{\mathbf{t}_{-i} \sim P(\cdot|t_i)}[U_i(a_i, \sigma_{-i}(\mathbf{t}_{-i}), t_i, \mathbf{t}_{-i})] \,.$$

Cognitive models of strategic play. Human behaviour in games deviates from BNE, particularly in one-shot games without prior experience. The **Cognitive Hierarchy** model (Camerer et al., 2004) proposes that players perform k levels of iterated reasoning with decreasing probability:

$$P(\text{player is level-}k) = e^{-\tau}\tau^k/k! \quad \text{(Poisson distribution with mean } \tau).$$

Level-0 players randomise uniformly. Level-k players best-respond to the distribution over lower-level

players. The mean level $\tau \approx 1.5$ fits a wide range of experimental data. Level-k models are formalised as a Bayesian model with a hierarchical prior over level, approximating the iterated application of the ToM operator k times.

0.175.3 25e.3 Cultural Evolution and Mathematical Models of Transmission

Cultural evolution models how beliefs, norms, and practices spread and change in populations over time. The **replicator dynamics** (Taylor and Jonker 1978; Hofbauer and Sigmund 1998) model cultural transmission as a selection process on a population of strategies. For a finite strategy set $\{1, \dots, k\}$ with frequencies $\mathbf{x} = (x_1, \dots, x_k)$ and payoff matrix A:

$$\dot{x}_i = x_i(f_i(\mathbf{x}) - \bar{f}(\mathbf{x})),$$

where $f_i(\mathbf{x}) = (A\mathbf{x})_i$ is the fitness of strategy i and $\bar{f}(\mathbf{x}) = \mathbf{x}^\top A\mathbf{x}$ is the mean fitness. Strategies above the mean frequency grow; strategies below the mean decline. Fixed points of the replicator dynamics correspond to Nash equilibria of the game A.

Iterated learning. The **iterated learning** framework (Kirby et al., 2015; Griffiths and Kalish 2007) models cultural evolution as a Markov chain over hypothesis spaces. At each generation, a learner observes data generated by the previous learner and infers a hypothesis $h \sim P(h|\text{data})$. The stationary distribution of this Markov chain is:

$$P_\infty(h) \propto P(h)^{1/\text{sample size}},$$

converging to the prior as sample size goes to infinity. This **inductive bias** result shows that iterated learning converges to the prior: cultural transmission over generations distils the inductive biases of learners into the stable cultural forms. Regularities in natural language (compositionality, systematicity, compressibility) reflect the inductive biases of human learners amplified over cultural transmission.

0.176 Appendix D: Mathematical Prerequisites and Background

This appendix collects standard mathematical background that is assumed throughout the series but not derived from scratch.

0.176.1 D.1 Linear Algebra

Let $\mathbf{A} \in \mathbb{R}^{m\times n}$. The **singular value decomposition** (SVD) is $\mathbf{A} = U\Sigma V^\top$ where $U \in \mathbb{R}^{m\times m}$, $V \in \mathbb{R}^{n\times n}$ are orthogonal and $\Sigma \in \mathbb{R}^{m\times n}$ is diagonal with non-negative entries $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r \geq 0$ (the singular values, $r = \min(m, n)$). The **Moore-Penrose pseudoinverse** is $\mathbf{A}^+ = V\Sigma^+U^\top$ where Σ^+ replaces each non-zero σ_i by $1/\sigma_i$.

Matrix determinant lemma: $(A+UCV)^{-1} = A^{-1} - A^{-1}U(C^{-1}+VA^{-1}U)^{-1}VA^{-1}$. Used throughout for updating matrix inverses when low-rank modifications are made (e.g., Kalman filter updates, kernel matrix computations).

Trace identities: $\text{tr}(ABC) = \text{tr}(CAB) = \text{tr}(BCA)$ (cyclic property). $\partial\,\text{tr}(AX)/\partial X = A^\top$. $\partial \log\det X/\partial X = X^{-\top}$ (for positive definite X).

Eigenvalue perturbation: if $A = A_0 + \epsilon B$, the first-order perturbation of the k-th eigenvalue is $\lambda_k \approx \lambda_k^{(0)} + \epsilon \mathbf{v}_k^\top B\mathbf{v}_k$ (by the Hellmann-Feynman theorem), where $\mathbf{v}_k$ is the k-th eigenvector of A_0.

0.176.2 D.2 Measure Theory and Integration

A **measure space** $(\Omega, \mathcal{F}, \mu)$ consists of a set Ω, a σ-algebra $\mathcal{F}$, and a measure $\mu : \mathcal{F} \to [0, \infty]$ satisfying $\mu(\emptyset) = 0$ and countable additivity. The **Lebesgue integral** $\int f \, d\mu$ generalises the Riemann integral to measurable functions on arbitrary measure spaces.

Radon-Nikodym theorem: if $\nu \ll \mu$ (i.e., $\mu(A) = 0 \Rightarrow \nu(A) = 0$), there exists a measurable function $f = d\nu/d\mu$ (the Radon-Nikodym derivative) such that $\nu(A) = \int_A f \, d\mu$ for all $A \in \mathcal{F}$.

Fubini-Tonelli: for σ-finite measures, $\int\int f \, d\mu \, d\nu = \int\int f \, d\nu \, d\mu$ (interchangeability of iterated integrals) when f is non-negative or $\int |f| \, d(\mu \otimes \nu) < \infty$.

0.176.3 D.3 Functional Analysis

A **Hilbert space** $\mathcal{H}$ is a complete inner product space. The **Riesz representation theorem**: for every bounded linear functional $\ell : \mathcal{H} \to \mathbb{R}$, there exists a unique $h_\ell \in \mathcal{H}$ such that $\ell(f) = \langle f, h_\ell \rangle$ for all f. This is the foundation of the reproducing kernel Hilbert space: the evaluation functional $\ell_x(f) = f(x)$ is bounded iff $\mathcal{H}$ has a reproducing kernel k, with $k(\cdot, x) = h_{\ell_x}$.

Spectral theorem: a compact self-adjoint operator $T : \mathcal{H} \to \mathcal{H}$ has a complete orthonormal basis of eigenvectors $\{e_j\}$ with eigenvalues $\lambda_j \to 0$, and $T = \sum_j \lambda_j e_j \otimes e_j$ (spectral decomposition). Applied to integral operators $T_k(f)(x) = \int k(x, y) f(y) \, dy$, this gives Mercer's theorem.

0.176.4 D.4 Differential Geometry

A **Riemannian manifold** (M, g) is a smooth manifold M equipped with a smoothly varying inner product g_p on each tangent space T_pM. The **Riemann curvature tensor** $R : TM^{\otimes 3} \to TM$ measures the non-commutativity of covariant derivatives: $R(X, Y)Z = \nabla_X \nabla_Y Z - \nabla_Y \nabla_X Z - \nabla_{[X,Y]} Z$. The **Ricci curvature** $\mathrm{Ric}(X, Y) = \mathrm{tr}[Z \mapsto R(Z, X)Y]$ and **scalar curvature** $R = \mathrm{tr}_g(\mathrm{Ric})$ are contractions.

For the statistical manifold with Fisher-Rao metric, the α-curvature tensors (Section 3c) characterise the geometry: exponential families are flat (e-flat), mixture families are flat (m-flat), and general families have non-zero curvature in both connections. The **Pythagorean theorem in information geometry**: for an e-flat submanifold $\mathcal{E}$ and an m-flat submanifold $\mathcal{M}$ with unique intersection point p^*, $\mathrm{KL}(q \| r) = \mathrm{KL}(q \| p^*) + \mathrm{KL}(p^* \| r)$ for all $q \in \mathcal{M}$, $r \in \mathcal{E}$. This is used in the proof that the m-projection onto an exponential family minimises KL divergence (Section 5.1).

0.177 Appendix A: Core Theorems of the Series

We collect the central mathematical results established across the twelve Section s.

Theorem 1 (Dutch Book Coherence). An agent's probability assignments are coherent (no Dutch Book exists against them) if and only if they satisfy the Kolmogorov axioms. Consequently, rational degrees of belief must be probabilities. *Proof sketch:* Construct, for any violation of the axioms, a finite set of bets each of which the agent must accept at its stated odds but whose conjunction guarantees a loss.

Theorem 2 (ELBO-KL Duality). For any generative model $p(x, z)$ and variational distribution $q_\lambda(z)$:

$$\log p(x) = \mathcal{L}(\lambda) + \mathrm{KL}(q_\lambda \| p(\cdot \mid x)), \quad \mathcal{L}(\lambda) \leq \log p(x).$$

Consequently, maximising the ELBO is equivalent to minimising $\mathrm{KL}(q_\lambda \| p(\cdot \mid x))$. *Proof:* Direct expansion using the definition of KL divergence.

Theorem 3 (GP Posterior in Closed Form). For a GP prior $f \sim \mathcal{GP}(0, k)$ and Gaussian likelihood $y_i = f(x_i) + \epsilon_i$, $\epsilon_i \sim \mathcal{N}(0, \sigma_n^2)$, the posterior at test inputs X_* is:

$$\mathbf{f}_* \mid X, \mathbf{y}, X_* \sim \mathcal{N}(K_{X_*X}(K_{XX} + \sigma_n^2 I)^{-1}\mathbf{y},\ K_{X_*X_*} - K_{X_*X}(K_{XX} + \sigma_n^2 I)^{-1}K_{XX_*}).$$

Proof: Standard conditioning of a jointly Gaussian vector.

Theorem 4 (Free-Energy Bound on Surprise). $\mathcal{F}(q, o) \geq -\log p(o)$ for any q, with equality iff $q = p(\cdot \mid o)$. Minimising free energy over q simultaneously minimises surprise and minimises KL from the recognition distribution to the true posterior. *Proof:* Follows from the non-negativity of KL divergence.

Theorem 5 (Grounding Indeterminacy). For any computable cognitive model F and any interpretation $\mathcal{I}$ consistent with F's observable input-output behaviour, there exist uncountably many interpretations $\mathcal{I}'$ consistent with the same behaviour but assigning different meanings to some symbols. *Proof:* Construct proxy functions on the symbol domain that preserve all observable structure; the set of such functions is (generically) uncountably infinite.

Theorem 6 (Functor Compositionality in DisCoCat). There exists a monoidal functor $F : \mathrm{Syn} \to$ **FdVect** from the pregroup grammar category to finite-dimensional vector spaces that is compositional: $F(\alpha \cdot \beta) = F(\alpha) \otimes F(\beta)$ for all syntactic types α, β, and $F(\text{derivation}) = \text{contraction of the corresponding tensors}$. *Proof:* Explicit construction of the functor on generators and morphisms.

These six theorems collectively define the mathematical landscape of the series: from the logic of rational belief (Theorem 1) through the algebra of approximate inference (Theorem 2), the analysis of function-space priors (Theorem 3), the dynamics of self-organising systems (Theorem 4), to the limits of formal approaches (Theorem 5) and the algebraic structure of compositionality (Theorem 6).

0.178 Appendix B: Connections Across the Series

Section **I** → Section **s II, IV, VII.** The Bayesian framework of Section I is the foundation for variational inference (Section II), the free-energy principle (Section IV), and probabilistic graphical models (Section VII). Bayes' theorem is not merely one tool among many; it is the structural backbone of the series.

Section **III** → Section **X.** Gaussian processes formalise the prior over continuous functions; ACT-R's base-level learning equation is an approximation to a GP prior over retrieval times. The connection is that both GP inference and rational memory are optimal Bayesian estimators under appropriate assumptions about the statistical structure of the environment.

Section **IV** → Section **X.** The free-energy principle (Section IV) and the predictive processing architecture (Section X) are related as theory to implementation: Active Inference specifies what is minimised; predictive processing specifies how the minimisation is implemented in a cortical hierarchy.

Section **V** → Section **s I–IV, VI.** IIT presupposes a causal model (Section VII), information-theoretic tools (Section VI), and the framework of Bayesian inference for characterising the distribution of states (Section I). Its distinctiveness lies not in its ingredients but in the specific use of intrinsic causal power,

defined by the Wasserstein distance between cut and intact cause-effect structures, as the identity of consciousness.

Section **VIII** → Section **XI.** Non-classical logics (Section VIII) and Montague semantics (Section XI) address the same problem from opposite directions: defeasible reasoning requires going beyond classical logic, while compositional semantics exploits the full power of the typed lambda calculus. The connection is through type theory: the Curry-Howard isomorphism identifies logical proofs with typed lambda terms, linking inference and meaning at a deep mathematical level.

Section **IX** → Section **s I–IV, VII.** Probabilistic programming (Section IX) is the synthesis of the series: it provides a language in which all the probabilistic models of Section s I–VII can be expressed as programs and subjected to unified inference algorithms. The Church language can express the generative models of Bayesian cognition, the causal models of Section VII, and the hierarchical models of Section I in a single computational framework.

Section **XII** → **all** Section **s.** The limits of Section XII are the necessary shadow of the positive results of all earlier Section s. Every formal framework examined in this series runs, at some point, into the grounding problem, the consciousness gap, the approximation barrier, or the model-misspecification wall. These limits do not invalidate the frameworks; they define the territory in which rigorous progress is possible and the frontiers that remain open.

0.179 Appendix C: Glossary of Mathematical Terms

Active Inference. The algorithmic implementation of the Free-Energy Principle, in which perception, action, and learning are all instances of free-energy minimisation with respect to a generative model. Introduced by Friston (2010).

ELBO (Evidence Lower BOund). The quantity $\mathcal{L}(\lambda) = \mathbb{E}_{q_\lambda}[\log p(x, z)] - \mathbb{E}_{q_\lambda}[\log q_\lambda(z)]$, which lower-bounds the log marginal likelihood $\log p(x)$. Central to variational inference and the VAE.

Free Energy (Variational). $\mathcal{F}(q, o) = \mathbb{E}_{q(s)}[\log q(s) - \log p(o, s)]$. An upper bound on the surprise $-\log p(o)$. Minimised by the recognition distribution $q^*(s) = p(s \mid o)$. The objective function of Active Inference.

Gaussian Process. A stochastic process $\{f(x) : x \in \mathcal{X}\}$ whose finite-dimensional marginals are all Gaussian. Specified by a mean function $m(x)$ and a covariance (kernel) function $k(x, x')$. Used as a non-parametric prior over functions.

Integrated Information (Φ). The system-level measure of integrated information in IIT: the Wasserstein-1 distance between the cause-effect structure of the intact system and the minimum-information partition. Proposed by Tononi as identical to the degree of consciousness.

KL Divergence. $\mathrm{KL}(q\|p) = \mathbb{E}_q[\log(q/p)] \geq 0$. Measures the extra coding cost of using distribution p to encode samples from q. Not symmetric: $\mathrm{KL}(q\|p) \neq \mathrm{KL}(p\|q)$ in general. Central to Bayesian inference and information theory.

Monoidal Category. A category equipped with a tensor product bifunctor $\otimes$ and unit object I, satisfying associativity and unit laws up to coherent natural isomorphism. The abstract setting for string diagram calculi and categorical compositionality.

Normalising Flow. A sequence of invertible differentiable transformations applied to a base distribution to produce a more complex distribution. The transformed density is computable via the change-of-variables formula and Jacobian determinants.

Pregroup Grammar. A grammar formalism based on compact closed categories (Lambek 2008). Types are elements of a pregroup; grammaticality of a string is determined by whether its type sequence reduces to the sentence type s using the pregroup reductions. Used in DisCoCat to couple syntax to semantics via a monoidal functor.

Predictive Coding. A hierarchical neural implementation of variational inference in which top-down signals carry predictions, bottom-up signals carry precision-weighted prediction errors, and inference is gradient descent on the variational free energy. Proposed by Rao and Ballard (1999), extended by Friston (2005).

Sensorimotor Contingency. The regular relationship between an agent's actions and the resulting sensory changes, formalised as a function from agent-environment configurations to tangent vectors on the sensory manifold. Proposed by O'Regan and Noë (2001) as the basis of perceptual experience.

Wasserstein Distance (W_p). The p-th order Wasserstein distance between probability measures μ and ν: $W_p(\mu,\nu) = (\inf_\gamma \int \|x-y\|^p \, d\gamma(x,y))^{1/p}$ where the infimum is over all couplings γ of μ and ν. Used in IIT to measure distance between cause-effect structures.

0.180 28. Continual Learning and Catastrophic Forgetting

0.180.1 28.1 The Stability-Plasticity Dilemma

A fundamental challenge for any learning system — biological or artificial — is to acquire new knowledge without obliterating what has already been learned. This tension is known as the **stability-plasticity dilemma** (Grossberg 1987): a system that is too plastic will overwrite old memories when learning new tasks (catastrophic forgetting), while a system that is too stable cannot adapt to new information.

Formally, consider a parametric model f_θ trained sequentially on tasks $\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_n$. After training on $\mathcal{T}_k$, the model has parameters θ_k. Catastrophic forgetting occurs when

$$\mathcal{L}_{\mathcal{T}_i}(\theta_k) \gg \mathcal{L}_{\mathcal{T}_i}(\theta_i), \quad i < k,$$

i.e., performance on old tasks degrades after training on new ones. In neural networks, this happens because gradient descent on $\mathcal{L}_{\mathcal{T}_k}$ moves parameters in directions that improve performance on $\mathcal{T}_k$ but may be arbitrary with respect to $\mathcal{T}_i$.

0.180.2 28.2 Elastic Weight Consolidation

Elastic Weight Consolidation (EWC; Kirkpatrick et al., 2017) is the most influential Bayesian approach to continual learning. The key insight is to treat the parameters θ as random variables and use the posterior from previous tasks as a prior for the new task.

After training on task $\mathcal{T}_1$, the posterior is $P(\theta|\mathcal{D}_1) \propto P(\mathcal{D}_1|\theta)P(\theta)$. When learning task $\mathcal{T}_2$, the Bayesian update is:

$$P(\theta|\mathcal{D}_1, \mathcal{D}_2) \propto P(\mathcal{D}_2|\theta)P(\theta|\mathcal{D}_1).$$

EWC approximates $P(\theta|\mathcal{D}_1)$ as a Gaussian centred at θ_1^* with precision equal to the **Fisher information matrix** F:

$$P(\theta|\mathcal{D}_1) \approx \mathcal{N}(\theta; \theta_1^*, F^{-1}), \quad F_{ij} = \mathbb{E}\left[\frac{\partial \log P(\mathcal{D}_1|\theta)}{\partial \theta_i}\frac{\partial \log P(\mathcal{D}_1|\theta)}{\partial \theta_j}\right]_{\theta=\theta_1^*}.$$

Training on $\mathcal{T}_2$ then optimises:

$$\mathcal{L}_{\text{EWC}}(\theta) = \mathcal{L}_{\mathcal{T}_2}(\theta) + \frac{\lambda}{2}\sum_i F_{ii}(\theta_i - \theta_{1,i}^*)^2.$$

The quadratic penalty anchors parameters that are important for $\mathcal{T}_1$ (high F_{ii}) near their old values, while allowing unimportant parameters to move freely.

The Fisher information diagonal F_{ii} quantifies the **importance** of parameter θ_i for task $\mathcal{T}_1$: a large F_{ii} means small changes to θ_i strongly affect the likelihood, so that parameter must be protected. EWC can be seen as a diagonal approximation to natural gradient descent with a Gaussian prior.

0.180.3 28.3 Progressive Networks and Architecture-Based Approaches

Progressive Neural Networks (Rusu et al., 2016) sidestep catastrophic forgetting by never modifying learned parameters: each new task gets its own column of layers, with lateral connections from all previous columns. Formally, the activation of unit j in layer ℓ for task k is:

$$h_j^{(\ell,k)} = \sigma\left(W^{(\ell,k)}h^{(\ell-1,k)} + \sum_{i<k} U^{(\ell,k:i)}h^{(\ell-1,i)}\right),$$

where $W^{(\ell,k)}$ is the weight matrix for task k and $U^{(\ell,k:i)}$ are lateral connection matrices from column i to column k. Previous columns are frozen; the new column can exploit previously learned features via the lateral connections.

0.180.4 28.4 Hippocampal Replay and Complementary Learning Systems

The neuroscientific solution to the stability-plasticity dilemma is the **Complementary Learning Systems** (CLS) theory (McClelland et al., 1995; Kumaran et al., 2016). CLS proposes two interacting memory systems:

- The **hippocampus** as a fast-learning, pattern-separated episodic memory. Individual experiences are stored with minimal overlap (sparse, orthogonal representations), enabling one-shot learning without interference.
- The **neocortex** as a slow-learning, distributed semantic memory. Knowledge is gradually consolidated from hippocampal replay, enabling generalisation and abstraction.

The mathematical model: let $\mathcal{M}_H$ be the hippocampal memory (a set of stored episodes) and f_{θ_C} be the cortical model. During sleep, the hippocampus replays episodes $(\mathbf{x}, y) \sim \mathcal{M}_H$, and the cortical model is updated by gradient descent on the replayed data:

$$\theta_C \leftarrow \theta_C - \eta\nabla_{\theta_C}\mathcal{L}(f_{\theta_C}(\mathbf{x}), y).$$

This **experience replay** mechanism was imported into deep RL by the DQN algorithm (Section X), where a replay buffer stores past transitions and the Q-network is updated from random minibatches. The mathematical connection to CLS: replay breaks temporal correlations in the training signal, enabling stable gradient descent on non-stationary data distributions.

Generative replay (Shin et al., 2017) extends this idea to continual learning without storing all past data: a generative model G (e.g., a VAE or GAN) is trained jointly with the task model. When learning a new task, pseudo-episodes are generated by G and used to rehearse previous tasks, preventing forgetting without access to the original data.

0.181 29. Neuro-Symbolic Integration: Bridging Continuous and Discrete Cognition

0.181.1 29.1 The Integration Problem

Neural networks excel at pattern recognition, robustness to noise, and statistical generalisation from data. Symbolic systems excel at systematic compositionality, interpretability, causal reasoning, and data efficiency. Neither paradigm alone captures the full spectrum of cognitive competence. **Neuro-symbolic integration** seeks a formal framework in which both coexist.

The gap can be stated precisely. Let $\mathcal{L}$ be a formal language with a set of sentences $\mathcal{S}$ and a model-theoretic semantics $[\![\cdot]\!] : \mathcal{S} \to \{\top, \bot\}$. Neural networks compute functions $f : \mathbb{R}^n \to \mathbb{R}^m$. The integration problem is to construct a mapping $\Phi : \mathbb{R}^n \to \mathcal{S}$ (from continuous representations to symbolic sentences) that is (i) differentiable, (ii) semantically faithful ($[\![\Phi(\mathbf{x})]\!]$ reflects the content of $\mathbf{x}$), and (iii) computationally tractable.

0.181.2 29.2 Probabilistic Soft Logic

Probabilistic Soft Logic (PSL; Bach et al., 2017) is a framework for reasoning with continuous-valued logical predicates. In PSL, each logical predicate $P(x)$ takes values in $[0, 1]$ rather than $\{0, 1\}$. Logical connectives are interpreted via the **Łukasiewicz t-norm**:

$$P \wedge Q = \max(0, P + Q - 1), \quad P \vee Q = \min(1, P + Q), \quad \neg P = 1 - P.$$

A PSL rule of the form $\forall x : A(x) \wedge B(x) \Rightarrow C(x)$ contributes a **hinge-loss** term to the energy:

$$E_r = \lambda_r \max(0,\ A(x) + B(x) - C(x) - 1)^p,$$

where $p \in \{1, 2\}$ and λ_r is the rule weight. The MAP solution is found by minimising the sum of energies over all rules and all groundings, a convex problem in $p = 2$. PSL allows domain knowledge expressed as logical rules to be combined with learned neural features, with gradients flowing through the soft truth values into the neural components.

0.181.3 29.3 Neural Theorem Proving

Neural Theorem Provers (NTP; Rocktäschel and Riedel, 2017) reformulate logic programming as a differentiable computation. The backward chaining algorithm of Prolog is rendered differentiable by

replacing symbol matching with **soft unification**: two symbols a and b unify with similarity $\sigma(a, b) = \exp(-\|\mathbf{e}_a - \mathbf{e}_b\|^2/\tau)$, where $\mathbf{e}_a, \mathbf{e}_b \in \mathbb{R}^d$ are learned embeddings and τ is a temperature. The proof score of a goal G given a knowledge base $\mathcal{KB}$ is computed by soft backward chaining:

$$\text{Prove}(G, \mathcal{KB}) = \max_{\text{proof trees}} \prod_{\text{unifications}} \sigma(\text{unification score}).$$

This is differentiable with respect to the symbol embeddings, allowing end-to-end training from examples of (question, answer) pairs. NTP learns both the symbolic rules and the embedding geometry that makes the rules applicable.

0.181.4 29.4 DeepProbLog and Probabilistic Logic Neural Networks

DeepProbLog (Manhaeve et al., 2018) extends ProbLog — a probabilistic logic programming language — with neural predicates. A neural predicate $nn(f, \mathbf{x}, y)$ states that the neural network f, when applied to input $\mathbf{x}$, assigns label y with the learned probability $P(y|\mathbf{x}; \theta_f)$. These neural predicate probabilities are used as annotated disjunctions in ProbLog programs, which are evaluated using the WMC (Weighted Model Counting) semiring. Gradients with respect to θ_f are computed through the WMC evaluation using the chain rule, enabling joint training of the neural and logical components.

Formally, a DeepProbLog program defines a probability distribution over Herbrand interpretations I:

$$P(\omega) = \prod_{\text{neural preds}} P(nn(\mathbf{x}, y; \theta))^{[nn(\mathbf{x},y) \in \omega]} \cdot P(\text{choices} \in \omega),$$

where the product ranges over all neural predicate groundings appearing in the least Herbrand model. The marginal probability of a query Q is $P(Q) = \sum_{\omega \models Q} P(\omega)$, computed by WMC. This architecture achieves state-of-the-art performance on tasks requiring both visual recognition (neural) and logical reasoning (symbolic), such as MNIST addition and visual question answering.

0.182 30. Mathematical Models of Emotion and Affective Cognition

0.182.1 30.1 Emotion as Interoceptive Inference

The mathematical treatment of emotion has historically lagged behind cognitive functions such as memory, attention, and language. The Active Inference framework (Section IV) offers a principled approach: **emotions are interoceptive inferences** — the brain's best guess about the causes of its own body states (Friston et al., 2010; Seth and Friston, 2016).

Formally, the body state is a hidden variable $\mathbf{b} \in \mathcal{B}$ that generates interoceptive sensations $\mathbf{s}_{\text{int}}$. The brain maintains a generative model $P(\mathbf{s}_{\text{int}}|\mathbf{b})P(\mathbf{b})$ and infers:

$$q^*(\mathbf{b}) = \arg\min_q \text{KL}(q(\mathbf{b}) \| P(\mathbf{b}|\mathbf{s}_{\text{int}})).$$

An **emotion** is the recognition density $q^*(\mathbf{b})$: the subjective feel of an emotional state is the inferred body state, not the body state itself. This resolves classical debates: the James-Lange theory (emotion is the perception of body changes) and the cognitive appraisal theory (emotion requires cognitive evaluation) are unified — emotion is a perception of the body, but that perception is mediated by a generative model

that incorporates prior expectations (appraisals).

0.182.2 30.2 Dimensional Models: Valence and Arousal

The dimensional approach to emotion represents affective states in a low-dimensional continuous space. Russell's **Circumplex Model** (1980) proposes two orthogonal dimensions:

- **Valence** $v \in [-1, +1]$: the hedonic quality of the emotion (pleasant to unpleasant).
- **Arousal** $a \in [0, 1]$: the activation level (calm to excited).

Emotions are points in the (v, a) plane. The geometry of the circumplex predicts that emotions equidistant from the origin should be equally intense, and that interpolation between emotions in the plane yields intermediate affective states. Psychophysiological evidence (facial electromyography, autonomic measures) supports the two-dimensional structure, with valence correlating with left/right frontal asymmetry and arousal correlating with sympathetic activation.

The mathematical formulation of the circumplex as a feature space: let $\phi(e) = (v(e), a(e)) \in \mathbb{R}^2$ be the embedding of emotion e. The similarity between emotions is $\langle \phi(e_1), \phi(e_2) \rangle / (\|\phi(e_1)\| \|\phi(e_2)\|)$ (cosine similarity in the affect space). Categorical emotion labels are recovered by Voronoi tessellation of the circumplex plane, connecting continuous and categorical descriptions.

0.182.3 30.3 Appraisal Theory: A Multi-Dimensional Computational Model

Appraisal theory (Scherer 1984; Ortony, Clore, Collins 1988) models emotion as the output of a cognitive evaluation process that assesses events along multiple dimensions (**appraisal dimensions**):

- **Relevance**: does this event matter to the agent's goals? $r \in [0, 1]$.
- **Implication**: is the outcome positive or negative for the agent? $\iota \in [-1, +1]$.
- **Coping potential**: can the agent change the outcome? $\kappa \in [0, 1]$.
- **Certainty**: how predictable is the situation? $c \in [0, 1]$.
- **Norm compatibility**: does the event violate social norms? $\nu \in [-1, +1]$.

The appraisal vector $\mathbf{a} = (r, \iota, \kappa, c, \nu)$ maps to an emotion category via a learned classifier or a rule-based system. For example: high relevance, negative implication, low coping potential, and low certainty produce *fear*; high relevance, negative implication, norm violation produce *anger*; positive implication with high coping potential produces *joy*.

Computationally, the OCC model (Ortony et al., 1988) specifies the appraisal rules as a decision tree over 22 emotion types. Modern implementations represent appraisals as probability distributions $P(\mathbf{a})$ (since events are often ambiguous in valence and relevance), propagate uncertainty through the appraisal rules, and produce categorical emotion probabilities $P(\text{emotion}|\mathbf{a})$, enabling reasoning about mixed and ambiguous affective states.

0.182.4 30.4 Emotion Regulation as Optimal Control

Emotion regulation refers to processes by which agents modulate their emotional states. Mathematically, this can be formalised as an optimal control problem over the affective state space.

Let $x_t \in \mathbb{R}^2$ be the valence-arousal state at time t, governed by the stochastic differential equation:

$$dx_t = f(x_t, u_t)\,dt + \sigma\,dW_t,$$

where $u_t \in \mathcal{U}$ is the regulation action (e.g., cognitive reappraisal, attentional deployment, behavioural avoidance), f is the drift (how the emotion evolves given the current state and action), and W_t is a Brownian motion capturing stochastic fluctuations in affect.

The agent seeks to minimise a cost functional:

$$J(u) = \mathbb{E}\left[\int_0^T \ell(x_t, u_t)\,dt + \Psi(x_T)\right],$$

where $\ell(x, u)$ is the running cost (e.g., penalising negative valence and high arousal) and Ψ is a terminal cost. The optimal control satisfies the **Hamilton-Jacobi-Bellman** (HJB) equation:

$$-\frac{\partial V}{\partial t} = \min_u \left[\ell(x, u) + \nabla_x V \cdot f(x, u) + \frac{\sigma^2}{2}\Delta_x V\right],$$

where $V(x, t)$ is the value function (minimum cost-to-go). Cognitive reappraisal corresponds to choosing u to shift f towards positive valence attractors; suppression corresponds to actions that reduce σ (damping affective fluctuations) at the cost of high u-cost.

0.183 31. AI Alignment: Mathematical Foundations

0.183.1 31.1 The Alignment Problem Stated Formally

The **AI alignment problem** asks: how can we ensure that an AI system with high capability pursues goals that are beneficial to humans? This is not merely an engineering question but a mathematical one: what is the formal structure of the problem, and what are the theoretical constraints on possible solutions?

Let an agent be defined by a utility function $U : \mathcal{S} \to \mathbb{R}$ over world states $\mathcal{S}$, a policy $\pi : \mathcal{H} \to \Delta(\mathcal{A})$ from histories to action distributions, and a world model $\mathcal{M}$. A **principal** (a human or society) has a utility function $U^* : \mathcal{S} \to \mathbb{R}$, which is typically **not** directly observable by the agent. Alignment requires $U \approx U^*$, but:

- U^* is not known to the agent in advance (it must be learned or inferred).
- U^* may be inconsistent, context-dependent, and difficult for the principal to articulate.
- A sufficiently capable agent optimising $U \neq U^*$ can cause catastrophic outcomes via **Goodhart's Law**: when a measure becomes a target, it ceases to be a good measure.

0.183.2 31.2 Inverse Reinforcement Learning

Inverse Reinforcement Learning (IRL; Ng and Russell, 2000) addresses the problem of inferring a reward function $R^* \in \mathcal{R}$ from observed expert behaviour π_E. Given:

- A Markov Decision Process $(\mathcal{S}, \mathcal{A}, P, \gamma)$ without the reward function,
- A set of expert trajectories $\tau_E = \{(s_0, a_0, s_1, a_1, \ldots)\}$ from policy π_E,

IRL seeks R^* such that $\pi_E = \arg\max_\pi V^\pi_{R^*}$, where $V^\pi_R = \mathbb{E}_\pi[\sum_t \gamma^t R(s_t)]$ is the value of policy π under reward R.

The IRL problem is fundamentally **ill-posed**: many reward functions are consistent with any observed policy. The degenerate solution $R \equiv 0$ always works. Disambiguation requires additional constraints, such as the **maximum entropy** principle (Ziebart et al., 2008): among all reward functions consistent with the observed behaviour, choose the one whose induced distribution over trajectories has maximum entropy:

$$R^* = \arg\max_R \mathrm{H}(P_R(\tau)) \;\text{ s.t. }\; \mathbb{E}_{P_R}[\phi(\tau)] = \mathbb{E}_{\pi_E}[\phi(\tau)],$$

where $\phi(\tau)$ is a feature vector of the trajectory. The maximum entropy IRL solution is $P_R(\tau) \propto \exp(\sum_t R(s_t, a_t))$, a Boltzmann distribution over trajectories, connecting IRL to probabilistic graphical models.

0.183.3 31.3 Cooperative AI and Principal-Agent Problems

Cooperative AI (Russell, 2019) reconceptualises alignment as a problem of **rational cooperation under uncertainty**. The agent does not optimise a fixed U but instead maintains uncertainty over the principal's utility function U^* and defers to human correction:

$$U_{\text{agent}} = \mathbb{E}_{P(U^*|\text{observations})}[U^*],$$

where the expectation is over the agent's posterior over U^*. Critically, the agent should assign positive probability to $U^* \neq U$ for any U it currently holds (epistemic humility about the utility function). This leads to three desirable behavioural properties proven by Russell (2019):

1. **Corrigibility:** the agent allows the principal to switch it off, because the agent is uncertain whether its current behaviour is optimal for U^* and values the principal's ability to correct it.
2. **Deference:** the agent prefers to query the principal rather than act unilaterally when uncertain, because principal queries have positive information value for the estimation of U^*.
3. **Convergence:** as the agent gathers more evidence about U^*, its behaviour converges to the principal's optimal policy.

The formal setting is a **cooperative game** $(U^*, U_{\text{agent}}, \mathcal{M})$ where agent and principal jointly solve a POMDP with hidden state U^*. The information-theoretic cost of misalignment is $\mathbb{E}[U^* - U^*(\pi_{\text{agent}})] = \mathbb{E}_{U^*}[V^*_{U^*} - V^{\pi_{\text{agent}}}_{U^*}]$, the expected regret under the true utility function.

0.183.4 31.4 Reward Modelling and Scalable Oversight

For tasks where human evaluation is expensive or infeasible, **reward modelling** trains a surrogate reward function $\hat{R}_\phi$ from human preference data. A preference dataset $\mathcal{D} = \{(\tau^+, \tau^-)\}$ consists of trajectory pairs where the human judged τ^+ preferable to τ^-. The Bradley-Terry model gives:

$$P(\tau^+ \succ \tau^-) = \sigma\left(\sum_t \hat{R}_\phi(s^+_t, a^+_t) - \sum_t \hat{R}_\phi(s^-_t, a^-_t)\right),$$

where σ is the sigmoid. Maximum likelihood estimation of ϕ gives the **reward model** used in Reinforcement Learning from Human Feedback (RLHF; Christiano et al., 2017). The policy is then trained

to maximise $\hat{R}_\phi$ with a KL penalty to prevent excessive deviation from a reference policy π_{ref}:

$$\max_{\pi} \mathbb{E}_\pi[\hat{R}_\phi(s,a)] - \beta\,\mathrm{KL}(\pi \| \pi_{\text{ref}}).$$

The mathematical risk is **reward hacking**: if $\hat{R}_\phi \neq R^*$, the policy may exploit the gap between $\hat{R}_\phi$ and R^* to score high on the learned reward model while violating the true objective. This connects to Goodhart's Law: $\hat{R}_\phi$ is a measure of R^* built from finite preference data, and optimising it aggressively will cause it to diverge from R^* wherever the preference data provides insufficient coverage.

0.183.5 31.5 Formal Verification and Specification

A complementary approach to alignment is **formal verification**: proving, before deployment, that an agent satisfies a formal safety specification φ expressed in a temporal logic (e.g., Linear Temporal Logic, LTL). A specification φ defines a set of acceptable trajectories $\mathcal{T}_\varphi = \{\tau : \tau \models \varphi\}$. Verification asks: does π_{agent} satisfy φ for all possible environment behaviours?

For finite-state MDPs, this reduces to model checking (polynomial in the size of the state space). For continuous state spaces and neural network policies, verification requires **neural network verification** tools (e.g., α-β-CROWN, Marabou), which use interval arithmetic and branch-and-bound search to certify that $f_\theta(\mathbf{x}) \in \mathcal{S}_{\text{safe}}$ for all $\mathbf{x}$ in a specified region. The computational complexity is NP-hard in general (the problem is as hard as satisfiability for piecewise-linear networks), but tractable for networks of moderate size with ReLU activations under tight input constraints.

The fundamental tension: formal specifications must be complete and unambiguous, but human values are neither. The mathematical challenge is to connect the partial, inconsistent, and informal specifications that humans can articulate with the complete, formal specifications required for verification — a problem that remains largely open and connects back to the grounding problem of Section XII.

Bibliography

Foundational Probability and Bayesian Inference

- Cox, R.T. (1946). Probability, Frequency and Reasonable Expectation. *American Journal of Physics*, 14(1), 1–13.
- de Finetti, B. (1937). La Prévision: Ses Lois Logiques, Ses Sources Subjectives. *Annales de l'Institut Henri Poincaré*, 7, 1–68.
- Jaynes, E.T. (2003). *Probability Theory: The Logic of Science*. Cambridge University Press.
- Kolmogorov, A.N. (1933). *Grundbegriffe der Wahrscheinlichkeitsrechnung*. Springer. [Engl. trans.: *Foundations of the Theory of Probability*, Chelsea, 1950.]
- MacKay, D.J.C. (2003). *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press.
- Savage, L.J. (1954). *The Foundations of Statistics*. Wiley.

Variational Inference and Approximate Bayesian Computation

- Blei, D.M., Kucukelbir, A., & McAuliffe, J.D. (2017). Variational Inference: A Review for Statisticians. *Journal of the American Statistical Association*, 112(518), 859–877.
- Jordan, M.I., Ghahramani, Z., Jaakkola, T.S., & Saul, L.K. (1999). An Introduction to Variational Methods for Graphical Models. *Machine Learning*, 37, 183–233.
- Kingma, D.P., & Welling, M. (2014). Auto-Encoding Variational Bayes. *International Conference on Learning Representations (ICLR)*.
- Rezende, D.J., Mohamed, S., & Wierstra, D. (2014). Stochastic Backpropagation and Approximate Inference in Deep Generative Models. *ICML 2014*.
- Wainwright, M.J., & Jordan, M.I. (2008). Graphical Models, Exponential Families, and Variational Inference. *Foundations and Trends in Machine Learning*, 1(1–2), 1–305.

Information Geometry and Differential Geometry

- Amari, S. (1985). *Differential-Geometrical Methods in Statistics*. Springer.
- Amari, S., & Nagaoka, H. (2000). *Methods of Information Geometry*. AMS / Oxford University Press.
- Cover, T.M., & Thomas, J.A. (2006). *Elements of Information Theory* (2nd ed.). Wiley.
- Lee, J.M. (2013). *Introduction to Smooth Manifolds* (2nd ed.). Springer.
- Shannon, C.E. (1948). A Mathematical Theory of Communication. *Bell System Technical Journal*, 27, 379–423, 623–656.

Gaussian Processes and Kernel Methods

- Rasmussen, C.E., & Williams, C.K.I. (2006). *Gaussian Processes for Machine Learning*. MIT Press.
- Schölkopf, B., & Smola, A.J. (2002). *Learning with Kernels*. MIT Press.
- Wahba, G. (1990). *Spline Models for Observational Data*. SIAM.

Free Energy Principle and Active Inference

- Friston, K. (2005). A Theory of Cortical Responses. *Philosophical Transactions of the Royal Society B*, 360, 815–836.
- Friston, K. (2010). The Free-Energy Principle: A Unified Brain Theory? *Nature Reviews Neuroscience*, 11, 127–138.
- Friston, K., Kilner, J., & Harrison, L. (2006). A Free Energy Principle for the Brain. *Journal of Physiology-Paris*, 100, 70–87.
- Parr, T., Pezzulo, G., & Friston, K. (2022). *Active Inference: The Free Energy Principle in Mind, Brain, and Behaviour*. MIT Press.
- Rao, R.P.N., & Ballard, D.H. (1999). Predictive Coding in the Visual Cortex. *Nature Neuroscience*, 2, 79–87.

Integrated Information Theory and Consciousness

- Oizumi, M., Albantakis, L., & Tononi, G. (2014). From the Phenomenology to the Mechanisms of Consciousness: Integrated Information Theory 3.0. *PLoS Computational Biology*, 10(5), e1003588.
- Tononi, G. (2004). An Information Integration Theory of Consciousness. *BMC Neuroscience*, 5, 42.
- Tononi, G., Boly, M., Massimini, M., & Koch, C. (2016). Integrated Information Theory: From Consciousness to Its Physical Substrate. *Nature Reviews Neuroscience*, 17, 450–461.
- Villani, C. (2009). *Optimal Transport: Old and New*. Springer.

Neural Networks and Deep Learning

- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
- Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer Feedforward Networks are Universal Approximators. *Neural Networks*, 2(5), 359–366.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep Learning. *Nature*, 521, 436–444.
- Vaswani, A., Shazeer, N., Parmar, N., et al. (2017). Attention Is All You Need. *Advances in Neural Information Processing Systems (NeurIPS)*, 30.

Probabilistic Graphical Models

- Koller, D., & Friedman, N. (2009). *Probabilistic Graphical Models: Principles and Techniques*. MIT Press.
- Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann.
- Pearl, J. (2000). *Causality: Models, Reasoning, and Inference*. Cambridge University Press.

Category Theory and Categorical Semantics

- Awodey, S. (2010). *Category Theory* (2nd ed.). Oxford University Press.
- Clark, B., Coecke, B., & Sadrzadeh, M. (2008). A Compositional Distributional Model of Meaning. *Proceedings of the Second Symposium on Quantum Interaction*, 133–140.
- Coecke, B., Sadrzadeh, M., & Clark, S. (2010). Mathematical Foundations for a Compositional Distributional Model of Meaning. *Linguistic Analysis*, 36, 345–384.
- Lambek, J. (2008). From Word to Sentence. Polimetrica.
- Mac Lane, S. (1998). *Categories for the Working Mathematician* (2nd ed.). Springer.

Logic, Formal Reasoning, and Non-Classical Logics

- Dummett, M. (1977). *Elements of Intuitionism*. Oxford University Press.
- Gärdenfors, P. (1988). *Knowledge in Flux: Modeling the Dynamics of Epistemic States*. MIT Press.
- Priest, G. (2008). *An Introduction to Non-Classical Logic* (2nd ed.). Cambridge University Press.
- Reiter, R. (1980). A Logic for Default Reasoning. *Artificial Intelligence*, 13(1–2), 81–132.

Formal Semantics and Computational Linguistics

- Barwise, J., & Perry, J. (1983). *Situations and Attitudes*. MIT Press.
- Montague, R. (1973). The Proper Treatment of Quantification in Ordinary English. In K. Hintikka et al. (eds.), *Approaches to Natural Language*. Reidel.
- Partee, B.H., ter Meulen, A., & Wall, R.E. (1990). *Mathematical Methods in Linguistics*. Kluwer.

Cognitive Science and Mathematical Psychology

- Anderson, J.R., & Lebiere, C. (1998). *The Atomic Components of Thought*. Lawrence Erlbaum.
- Chalmers, D.J. (1996). *The Conscious Mind*. Oxford University Press.
- Harnad, S. (1990). The Symbol Grounding Problem. *Physica D*, 42, 335–346.
- O'Regan, J.K., & Noë, A. (2001). A Sensorimotor Account of Vision and Visual Consciousness. *Behavioral and Brain Sciences*, 24(5), 939–973.
- Tenenbaum, J.B., Kemp, C., Griffiths, T.L., & Goodman, N.D. (2011). How to Grow a Mind: Statistics, Structure, and Abstraction. *Science*, 331, 1279–1285.

Continual Learning and Catastrophic Forgetting

- Grossberg, S. (1987). Competitive Learning: From Interactive Activation to Adaptive Resonance. *Cognitive Science*, 11(1), 23–63.
- Kirkpatrick, J., Pascanu, R., Rabinowitz, N., et al. (2017). Overcoming Catastrophic Forgetting in Neural Networks. *Proceedings of the National Academy of Sciences*, 114(13), 3521–3526.
- Kumaran, D., Hassabis, D., & McClelland, J.L. (2016). What Learning Systems Do Intelligent Agents Need? Complementary Learning Systems Theory Updated. *Trends in Cognitive Sciences*, 20(7), 512–534.
- McClelland, J.L., McNaughton, B.L., & O'Reilly, R.C. (1995). Why There Are Complementary Learning Systems in the Hippocampus and Neocortex. *Psychological Review*, 102(3), 419–457.
- Parisi, G.I., Kemker, R., Part, J.L., Kanan, C., & Wermter, S. (2019). Continual Lifelong Learning with Neural Networks: A Review. *Neural Networks*, 113, 54–71.
- Rusu, A.A., Rabinowitz, N.C., Desjardins, G., et al. (2016). Progressive Neural Networks. *arXiv:1606.04671*.
- Shin, H., Lee, J.K., Kim, J., & Kim, J. (2017). Continual Learning with Deep Generative Replay. *Advances in Neural Information Processing Systems (NeurIPS)*, 30.
- Zenke, F., Poole, B., & Ganguli, S. (2017). Continual Learning Through Synaptic Intelligence. *ICML 2017*.

Neuro-Symbolic Integration

- Bach, S., Broecheler, M., Huang, B., & Getoor, L. (2017). Hinge-Loss Markov Random Fields and Probabilistic Soft Logic. *Journal of Machine Learning Research*, 18, 1–67.
- Garcez, A., Broda, K., & Gabbay, D. (2002). *Neural-Symbolic Learning Systems*. Springer.
- Garnelo, M., & Shanahan, M. (2019). Reconciling Deep Learning with Symbolic Artificial Intelligence. *Current Opinion in Behavioral Sciences*, 29, 17–23.
- Lake, B.M., Ullman, T.D., Tenenbaum, J.B., & Gershman, S.J. (2017). Building Machines That Learn and Think Like People. *Behavioral and Brain Sciences*, 40, e253.
- Manhaeve, R., Dumancic, S., Kimmig, A., Demeester, T., & De Raedt, L. (2018). DeepProbLog: Neural Probabilistic Logic Programming. *NeurIPS 2018*.
- Marcus, G. (2019). Rebooting AI. Pantheon Books.
- Rocktäschel, T., & Riedel, S. (2017). End-to-End Differentiable Proving. *NeurIPS 2017*.
- Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., & Hovy, E. (2016). Hierarchical Attention Networks for Document Classification. *NAACL-HLT 2016*.

Mathematical Models of Emotion and Affective Cognition

- Barrett, L.F. (2017). *How Emotions Are Made: The Secret Life of the Brain*. Houghton Mifflin Harcourt.

- Colombetti, G. (2014). *The Feeling Body: Affective Science Meets the Enactive Mind*. MIT Press.
- Gross, J.J. (1998). The Emerging Field of Emotion Regulation. *Review of General Psychology*, 2(3), 271–299.
- Ortony, A., Clore, G.L., & Collins, A. (1988). *The Cognitive Structure of Emotions*. Cambridge University Press.
- Russell, J.A. (1980). A Circumplex Model of Affect. *Journal of Personality and Social Psychology*, 39(6), 1161–1178.
- Scherer, K.R. (1984). On the Nature and Function of Emotion. In K.R. Scherer & P. Ekman (eds.), *Approaches to Emotion*. Lawrence Erlbaum.
- Seth, A.K., & Friston, K.J. (2016). Active Interoceptive Inference and the Emotional Brain. *Philosophical Transactions of the Royal Society B*, 371, 20160007.

AI Alignment: Mathematical Foundations

- Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., & Mané, D. (2016). Concrete Problems in AI Safety. *arXiv:1606.06565*.
- Bai, Y., Jones, A., Ndousse, K., et al. (2022). Training a Helpful and Harmless Assistant with Reinforcement Learning from Human Feedback. *arXiv:2204.05862*.
- Christiano, P., Leike, J., Brown, T.B., et al. (2017). Deep Reinforcement Learning from Human Preferences. *NeurIPS 2017*.
- Hadfield-Menell, D., Dragan, A., Abbeel, P., & Russell, S. (2016). Cooperative Inverse Reinforcement Learning. *NeurIPS 2016*.
- Leike, J., Martic, M., Krakovna, V., et al. (2018). AI Safety Gridworlds. *arXiv:1711.09883*.
- Ng, A.Y., & Russell, S. (2000). Algorithms for Inverse Reinforcement Learning. *ICML 2000*.
- Russell, S. (2019). *Human Compatible: Artificial Intelligence and the Problem of Control*. Viking.
- Soares, N., & Fallenstein, B. (2014). Aligning Superintelligence with Human Interests. *MIRI Technical Report 2014-8*.
- Ziebart, B.D., Maas, A., Bagnell, J.A., & Dey, A.K. (2008). Maximum Entropy Inverse Reinforcement Learning. *AAAI 2008*.

Computability, Complexity, and Limits of Formal Systems

- Gödel, K. (1931). Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I. *Monatshefte für Mathematik und Physik*, 38, 173–198.
- Li, M., & Vitányi, P. (2008). *An Introduction to Kolmogorov Complexity and Its Applications* (3rd ed.). Springer.
- Putnam, H. (1967). Psychological Predicates. In W.H. Capitan & D.D. Merrill (eds.), *Art, Mind, and Religion*. University of Pittsburgh Press.

- Sipser, M. (2013). *Introduction to the Theory of Computation* (3rd ed.). Cengage.
- Turing, A.M. (1936). On Computable Numbers, with an Application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, 42, 230–265.